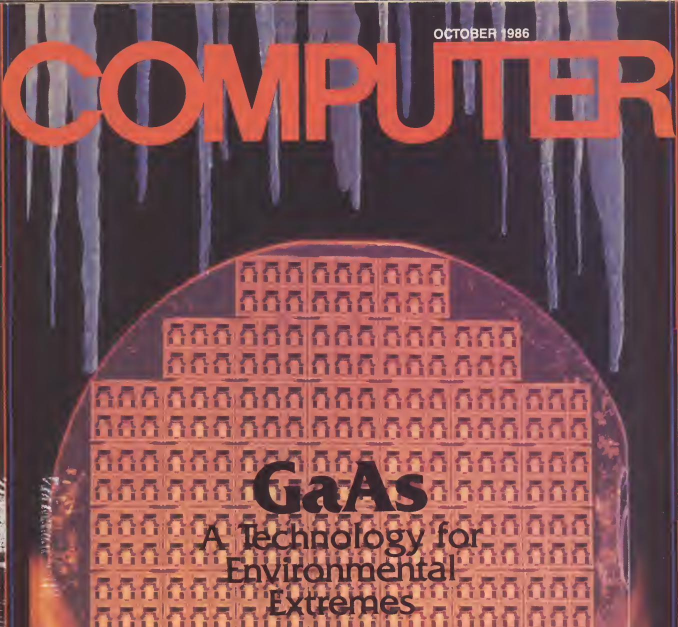
OCTOBER 1986

COMPUTER

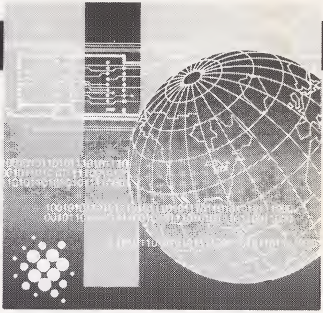


GaAs

A Technology for Environmental Extremes

IEEE COMPUTER SOCIETY

THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, INC.



Software Professionals:

LEAD THE EVOLUTION

At GTE Government Systems, we are creating an environment that will influence the evolution of software development into the next decade and beyond. If you have the high-level skills we seek, you can lead this evolution now through one of the following roles:

SOFTWARE ENVIRONMENT TECHNOLOGY LEADER

With Ada* as the basis for large-scale systems development, we are developing a methodology based on structured analysis and design employing an APSE architecture based on distributed VAX/VMS. Reusable software and software prototyping are key enhancements.

You will direct the planning and development of an evolving software environment; design a strategic APSE architecture; plan a methodology evolution; evaluate emerging software engineering technologies; and identify and develop new business opportunities in the software environment.

Requires experience designing systems using Ada and an active role in the software engineering community.

SOFTWARE SYSTEMS TECHNOLOGY LEADER

We are developing a next-generation architecture for digital voice processing systems based on the technologies of digital speech processing, high-bandwidth LANs, high-performance workstations, distributed computer security and data bases, and very large mass storage hierarchies.

We are evaluating emerging technologies including: expert systems applied to complex tasking problems, parallel processing in Ada for signal processing applications, and multi-level secure operating systems.

You will provide technical leadership in the development and evolution of architecture; lead IR&D tasks; evaluate emerging technologies; develop software architectures for new business opportunities; and review architectures from the technology and feasibility perspective.

Requires knowledge of technologies and their directions with experience in studies and proposals.

SOFTWARE PROJECT MANAGERS

Our strategic business plan targets substantial engineering growth taking advantage of our leadership in Ada-based software intensive systems.

You will assist in identifying and leading specific opportunities from marketing through management of the software project; create software development plans; develop competitive, realistic software costs; manage the software portion of proposals; and develop the organization to implement acquired projects.

Requires experience and insight into software project costing, planning, and control in a DoD contracting environment.

SOFTWARE TEST MANAGER

Our software test team's involvement spans the full range of test activities from initial design and analysis through system level testing. We use the latest automated tools and techniques for software test, tracking, and analysis.

You will be responsible for systems software integration and test requirements analysis; design analysis; ensuring testability of software design/development effort; determining and developing software test plans and procedures; and generating test data for a large, real-time software development project.

We offer competitive compensation, a professional work environment, and complete benefits including educational assistance, a stock purchase plan, a tax-deferred savings plan, and much more.

Our Washington office is located within minutes of the nation's capital, one of the world's major power and cultural centers. Send your resume in confidence to:

GTE Government Systems Corporation
Washington Operations
1700 Research Boulevard
Suite 200HK
Rockville, MD 20850

Similar positions are also available in Mountain View, CA, located in Silicon Valley near the beautiful San Francisco Bay area. Send your resume to:

GTE Government Systems Corporation
Western Division
P.O. Box 7188
Dept. CC367
100 Ferguson Drive
Mountain View, CA 94039

An equal opportunity employer. U.S. citizenship is required.

*Ada is a registered trademark of the U.S. Government, Ada Joint Program Office.

GTE Government Systems

SIMULATION NEWS

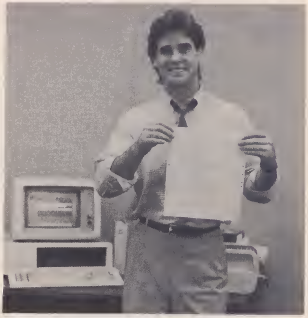
Now - SIMSCRIPT II.5 reduces the cost of simulation by 75%

You get results sooner and they are better understood

before

FORTRAN Code—4 times longer, 4 times costlier and unreadable

after



SIMSCRIPT II.5—simplifies and reduces costs

special offer -- free trial and training

SIMSCRIPT II.5 gives you a compact, English-like language that helps you succeed by saving your organization lots of money.

Your simulation program is 75% smaller and reads like a description of the system you are studying. **Your model development, checkout, modification and enhancement are greatly simplified.**

Widely used and fully supported

SIMSCRIPT II.5 is a well established, standardized, and widely used language with proven software support.

Typical applications include: military planning, manufacturing, communications, logistics, and transportation. These are complex systems and yet, you need to provide results in a timely manner. **With SIMSCRIPT II.5 you get results sooner and they are better understood.**

Computers with SIMSCRIPT II.5

- IBM AT, XT or compatible.
- Most mainframe computer types including IBM, CDC, VAX, Univac, Gould, Prime, Data General and ETA. Sun SIMSCRIPT is under development.

Free trial and special offer

The free trial contains everything you need to try SIMSCRIPT II.5® on your own computer.

We send you PC or Mainframe SIMSCRIPT, installation instructions, sample models, and a complete set of documentation. You can build your own model or modify one of ours. If you have questions, just call. **No cost or obligation.**

For a limited time we will also include free training. Act now to avoid disappointment. **For immediate information call Hal Duncan at (619) 457-9681.**

SIMSCRIPT II.5 is a registered trademark and service mark of CACI, INC.-FEDERAL

©1986 CACI, INC.-FEDERAL

Rush information on SIMSCRIPT II.5 special training offer and free trial

free trial-learn the reasons for the broad and growing popularity of SIMSCRIPT II.5. Try Mainframe or PC XT-AT SIMSCRIPT II.5, user instructions, training, documentation, and user support—no cost or obligation.

special offer-return the coupon today and you will be eligible for free training.

☐Also send a free copy of the publication: the ten most frequent causes of simulation analysis failure.

Name

Organization

Address

City State Zip

Telephone

Computer Operating System

IEEE COMP

Return to:
CACI
Products Company
3344 North Torrey Pines Court
La Jolla, California 92037
or better yet,
call Hal Duncan at (619) 457-9681

COMPUTER

October 1986 Volume 19 Number 10 (ISSN 0018-9162)

ARTICLES

10 Guest Editor's Introduction: GaAs Microprocessor Technology

Veljko Milutinović

GaAs will never totally replace silicon; it will be used in selected aerospace, defense, and supercomputing applications. The extent of its usage is open to question.

17 DARPA, SDI, and GaAs

Sherman Karp and *Sven Roosild*

DARPA and the SDI support the development of GaAs technology because of its vital military applications. It has great potential for high-speed signal processing, low power dissipation, and high radiation resistance.

21 GaAs High-Speed Digital IC Technology: An Overview

Lawrence E. Larson, *Joseph F. Jensen*, and *Paul T. Greiling*

GaAs IC technology has matured to the point of fabrication of large-scale integrated circuits. It has definite potential in high-speed digital systems in the near and long term.

29 Signal Processors Based Upon GaAs ICs: The Need for a Wholistic Design Approach

Barry K. Gilbert, *Barbara A. Naused*, *Daniel J. Schwab*, and *Rick L. Thompson*

Recasting traditional signal-processor architectures at the memory-layout, logic-design, arithmetic-implementation, and system-architecture levels will permit fuller use of GaAs technology.

45 Issues of Importance in Designing GaAs Microcomputer Systems

Veljko Milutinović, *Alex Silbey*, *David Fura*, *Kevin Keirn*, *Mark Bettinger*, *Walt Helbig*, *William Heagerty*, *Richard Zieger*, *Bob Schellack*, and *Walt Curtice*

Many more factors are proving to be drivers of GaAs VLSI architecture than was ever the case with silicon. However, the possibility that GaAs may make possible throughput five times greater than that achievable with silicon continues to encourage research.

60 A 32-bit RISC Implemented in Enhancement-Mode JFET GaAs

Terrence L. Rasset, *Roger A. Niederland*, *John H. Lane*, and *William A. Geideman*

This CPU architecture exemplifies the special design approaches being developed to take advantage of GaAs technology for high-speed processing.

71 Reduced Instruction Set Architecture for a GaAs Microprocessor System

Eric R. Fox, *Kenneth J. Kiefer*, *Robert F. Vangen*, and *Shaun P. Whalen*

RISC architecture characteristics of low hardware requirements, large register files, and pipelining accommodate the gate count constraints and preserve the inherent speed of GaAs technology.

Circulation: *Computer* (ISSN 0018-9162) is published monthly by the IEEE Computer Society: IEEE Headquarters—345 East 47th St., New York, NY 10017; IEEE Computer Society West Coast Office, 10662 Los Vaqueros Circle, Los Alamitos, CA 90720. Annual subscription included in society member dues. Nonmember prices: available upon request. Single copy prices: members $7.50; nonmembers $15.00. This journal is also available in microfiche form.

Undelivered copies: Send to 10662 Los Vaqueros Circle, Los Alamitos, CA 90720.

Postmaster: Send address changes to *Computer*, IEEE Service Center, 445 Hoes Lane, Piscataway, NJ 08854. Second class postage is paid at New York, New York, and at additional mailing offices.

Copyright and reprint permissions: Abstracting is permitted with credit to the source. Libraries are permitted to photocopy beyond the limits of US copyright law for private use of patrons: (1) those post-1977 articles that carry a code at the bottom of the first page, provided the per-copy fee indicated in the code is paid through the Copyright Clearance Center, 29 Congress Street, Salem, MA 01970. (2) pre-1978 articles without fee. Instructors are permitted to photocopy isolated articles for noncommercial classroom use without fee. For other copying, reprint, or republication permission, write to Editor, *Computer*, 10662 Los Vaqueros Circle, Los Alamitos, CA 90720. All rights reserved. Copyright © 1986 by The Institute of Electrical and Electronics Engineers, Inc.

Editorial: Unless otherwise stated, bylined articles, as well as products and services offered in New Products and New Applications, reflect the author's or firm's opinion; inclusion does not necessarily constitute endorsement by the IEEE or the Computer Society.

GaAs Microprocessor Technology

DEPARTMENTS

4 Letters to the Editor: June issue criticized

8 From the President: Board acts to cut expenses, increase revenues; Tom Cain now acting-president

82 Update: Silicon/GaAs research; McDowell award nominations; Committee on Public Policy 1986/87 plans; Computer chess championship; Erratum—September issue

85 Open Channel: The emperor's new angels; Which system is a RISC?

90 Calendar

92 Call for Papers

95 Book Reviews: Advanced Database Machine Architecture; Programming in micro-PROLOG; New Computer Architectures; Organization and Management of a Computer Center

98 New Products: A sampling of the GaAs market

102 New Literature

104 Microsystem Announcements

106 IC Announcements

107 Career Opportunities

112 Advertiser/Product Index

112A Reader Service Card

4 Change-of-Address Form

Editor-in-Chief: Michael C. Mulder, University of Portland

Editorial Board
Dharma P. Agrawal, North Carolina State University
Peter P. Chen, Louisiana State University
Amrit L. Goel, Syracuse University
Jim Haynes, University of California, Santa Cruz
Ronald G. Hoelzeman, University of Pittsburgh
Franklin F. Kuo, SRI International
Francis P. Mathur,
California State Polytechnic University, Pomona
Demetrios A. Michalopoulos,
California State University, Fullerton
K. M. Mohiuddin, IBM Corporation
Leon J. Osterweil, University of Colorado, Boulder
Edward A. Parrish, Jr., University of Virginia
Arthur V. Pohm, Iowa State University
Ralph Preiss, IBM Corporation
C. V. Ramamoorthy, University of California, Berkeley
Sallie Sheppard, Texas A&M University
Harry Strong, Mitre Corporation
Wing N. Toy, AT&T Bell Laboratories
Herbert Weber, University of Bremen
Helen M. Wood, National Bureau of Standards

Magazine Advisory Committee:
Dennis R. Allison (chair), Dharma P. Agrawal, Vishwani D. Agrawal,
James J. Farrell III, Lansing Hatfield, Michael C. Mulder, David Pessel,
True Seaborn, Bruce D. Shriver

Staff

Editor and Publisher: True Seaborn

Managing Editor: Marilyn Potes

Issue Editor: Louise Anderson

Assistant Editors: Nancy Hays, Thomas Szalkiewicz

Contributing Editor: Ware Myers

Assistant to the Publisher: Patricia Paulsen

Advertising Director: Michael Koehler

Advertising Coordinator: Carole Porter

Membership/Circulation Manager: Christina Champion

Staff Writer: Torrey Byles

Art Director: Jay Simpson

Production Supervisor: David Gaines

Design and Production: Miriam Wiegel

Submissions: Submit six copies of all articles and special-issue proposals to Michael C. Mulder, Director, Applied Research, University of Portland, Portland, OR 97203; (503) 283-7433.

On the cover

GaAs will never totally replace silicon, but it is expected to prove especially useful where extremes, whether of temperature or radiation, are encountered and where another extreme, its rapid switching speed, would be advantageous. Thus, it should find its greatest use in aerospace, defense, and supercomputing applications.

Cover elements: Icicles from a photo by Scott Blackman, Tom Stack & Associates; GaAs wafer courtesy of Vitesse Electronics Corporation; fire and special-effects photography by Larry Keiser.

Cover design: Jay Simpson

In the next issue

Trends and progress reports

IEEE COMPUTER SOCIETY
EXECUTIVE COMMITTEE

President: Roy L. Russo*

Acting-President: J. T. Cain*
University of Pittsburgh
Department of Electrical Engineering
Pittsburgh, PA 15261
(412) 624-5394

Vice Presidents
Publications (1st VP): J. T. Cain*
Technical Activities (2nd VP): John D. Musa*
Conferences and Tutorials: James H. Aylor
Educational Activities: Glen G. Langdon, Jr.
Membership and Information: Ming T. Liu
Area Activities: H. Troy Nagle
Standards: Helen M. Wood

Treasurer: Joseph E. Urban
Secretary: Fletcher J. Buckley
Junior Past President: Martha Sloan*
IEEE Division Directors: Martha Sloan,
Ronald G. Hoelzeman
Executive Director: T. Michael Elliott*
*Ex officio member of Board of Governors

BOARD OF GOVERNORS

TERM ENDING 1986	TERM ENDING 1987
Dennis R. Allison	Barry W. Boehm
Kenneth R. Anderson	Paul L. Borrill
P. Bruce Berra	Glen G. Langdon, Jr.
Fletcher J. Buckley	Duncan H. Lawrie
Richard C. Jaeger	Susan L. Rosenbaum
Ming T. Liu	Bruce Shriver
Michael C. Mulder	Harold S. Stone
Hillel Ofek	Wing N. Toy
Edward W. Thomas	Helen M. Wood
Joseph E. Urban	Akihiko Yamada
Oscar N. Garcia*	

PUBLICATIONS BOARD

Dharma P. Agrawal
Vishwani D. Agrawal
Dennis R. Allison
Bill D. Carroll
Michael Evangelist
James J. Farrell III
Tse-yun Feng
Lansing Hatfield
Ronald G. Hoelzeman
Sam Horvitz
Richard C. Jaeger
Willis K. King
Duncan H. Lawrie
Jack Lipovski
Ming T. Liu
Michael C. Mulder
Theo Pavlidis
David Pessel
C. V. Ramamoorthy
Bruce D. Shriver
Steve Tanimoto

J. T. Cain, Vice President for Publications

SENIOR STAFF

Executive Director: T. Michael Elliott
IEEE Computer Society
1730 Massachusetts Ave., NW
Washington, DC 20036-1903
(202) 371-0101

(Will furnish complete roster
of committees upon request.)

Editor and Publisher: True Seaborn
Director, Computer Society Press: Chip G. Stockton
Director, Conferences: William R. Habingreither
Director, Tutorials: Martez A. Camillieri
Director, Finance: Mary Ellen Curto

NEXT BOARD OF GOVERNORS MEETING:
Hotel Anatole
Dallas, Texas
November 7, 1986, 8:30 a.m.-5 p.m.

THE INSTITUTE OF ELECTRICAL
AND ELECTRONICS ENGINEERS, INC.

President: Bruno O. Weinschel
President-Elect: Henry L. Bachman
Executive Vice President: Emerson W. Pugh
Executive Director: Eric Herz

LETTERS

June issue criticized

To the editor:

The purpose of this letter is to provide a review of the quality of the articles in *Computer* magazine. In this sense, the aim is at the quality of the professional writing of the articles rather than at their technical content or appropriateness.

The June 1986 issue of *Computer* has just arrived, and, alas, its quality appears to be spotty. Specifically, the issue contains four articles that represent poor professional writing. This is a judgment that is based on the following criteria:

(1) Every article should have a statement of purpose in the first paragraph. This enables the reader to determine if he or she wants to continue to read the article. (To those who believe, no explanation is necessary. To those who do not believe, no explanation is possible.)

(2) Every paragraph should contain one and only one thought. This is normally expressed in the lead sentence. The remainder of the paragraph expands on this thought.

(3) Each sentence should contain not more than 25 words.

As examples of good practice, consider the first-paragraph statements of purpose in the articles by John Maier (''This article explores China's. . . '') and by Mark Sherman and Ann Marks (''Throughout this article we wish to share our experiences in designing. . . '').

As examples of poor practice, consider four other articles:

- The lead article, ''Education and America's Industrial Future.'' I have read it twice. I do not know what the author's purpose was in writing the article. I can speculate that it was a factual report, or that it was a call to arms and that something should be done (perhaps). Don't know. I am reluctant to read it a third time.
- The second article, ''Design Education in Computer Science and Engineering.'' Again, I am unable to divine its purpose. Is it to state what has been done, what should be done, what the need is, etc.?
- ''Vocationalism and the Whole Man'' and ''A Visual Approach to Browsing in a Database Environment.'' Previous comments apply.

Can we accept that we should have clear articles in *Computer*? If so, then can we accept that there should be written criteria against which a judgment can be made? Or is our position that our standards are that there are no standards and that academic freedom knows no bounds?

We, as members of the Computer Society, have this lack of quality foisted off on us every month. It is time for a change.

Fletcher J. Buckley
Cherry Hill, New Jersey

Moving?

PLEASE NOTIFY
US 4 WEEKS
IN ADVANCE

Name (Please Print)

New Address

City State/Country Zip

MAIL TO:
IEEE Service Center
445 Hoes Lane
Piscataway, NJ 08854

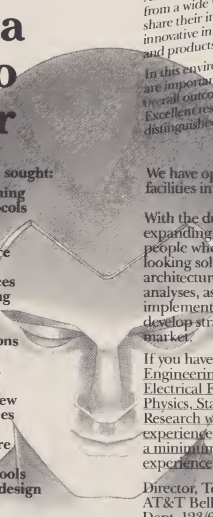
ATTACH
LABEL
HERE

- This notice of address change will apply to all IEEE publications to which you subscribe.
- List new address above.
- If you have a question about your subscription, place label here and clip this form to your letter.

Systems Engineers System Architects
Software Developers

AT&T Bell Laboratories

Where one idea leads to another



At AT&T Bell Laboratories talented individuals from a wide variety of technical backgrounds share their insights and knowledge to create innovative information systems, services and products.

In this environment, every person's contributions are important and personal ownership in the overall outcome of each effort is valued. Excellent resources are at hand to support distinguished results.

Specific expertises being sought:

- ☐ **Communications planning**
- ☐ **Communications protocols**
- ☐ **Computer networking**
- ☐ **Database management**
- ☐ **Design and manufacture integration planning**
- ☐ **Digital networks/services**
- ☐ **Digital signal processing**
- ☐ **Expert systems**
- ☐ **Human factors**
- ☐ **International applications**
- ☐ **Integrated networks**
- ☐ **Lightwave applications**
- ☐ **Local area networks**
- ☐ **Market definition for new technologies and services**
- ☐ **Measurement systems**
- ☐ **Microprocessor software development**
- ☐ **Network optimization tools**
- ☐ **Network planning and design**
- ☐ **Operations automation**
- ☐ **Packet switching**
- ☐ **Performance measurement and analysis**
- ☐ **Product planning**
- ☐ **Real time system software**
- ☐ **Software development tools**
- ☐ **Subjective testing**
- ☐ **Systems architecture**
- ☐ **Systems planning**
- ☐ **Transmission systems**
- ☐ **UNIX™ systems applications**

We have opportunities available at our facilities in New Jersey, Illinois and Ohio.

With the development of new and expanding projects, we are seeking people who can create forward-looking solutions, evaluate alternative architectures, perform competitive analyses, assess service viability, define, implement and measure systems or develop strategies to get products to market.

If you have a Master's or PhD in Systems Engineering, Computer Science, Electrical Engineering, Mathematics, Physics, Statistics, or Operations Research with hardware and/or software experience, or a BS in these fields with a minimum of 5 years highly relevant experience, please send your resume to:

Director, Technical Employment
AT&T Bell Laboratories
Dept. 123/6840/86
101 J.F. Kennedy Parkway
Short Hills, NJ 07078-0905

An equal opportunity employer

AT&T
The right choice.

Digital has it now.

With Digital's VAXstation II/GPX™ color workstation, Computer Aided Engineering and Computer Aided Software Engineering projects can now be tied into the computing resources of the entire company.

VAXstation II/GPX's built-in networking capabilities provide access to Digital's large VAX™ systems via Ethernet and DECnet™ networking software. And to other vendors' systems via gateways and other communications protocols. By off-loading compute-intensive tasks to larger computers, the MicroVAX II™ CPU and GPX graphics coprocessor can concentrate on delivering exceptional graphics at tremendous speeds. Plus, you can monitor all aspects of a project on the screen's multiple windows.

VAXstation II/GPX: The end of isolation for CAE and CASE applications.

VAXstation II/GPX runs popular CAE applications from such companies as Scientific Calculations, Inc., Silvar-Lisco,™ Tektronix™ – CAE Systems Division and VLSI Technology, Inc. It also runs CASE applications from B.S.O., Interactive Development Environments, Nastec Corp. and Tektronix – SDP Division.

VAXstation II/GPX. The workstation comes out of isolation and into the mainstream. For brochures, write: Digital Equipment Corporation, Media Response Manager, 200 Baker Ave., West Concord, MA 01742. Or call your local sales office.

digital™

© Digital Equipment Corporation 1986. Digital, the Digital logo, DECnet, MicroVAX II, VAX and VAXstation II/GPX are trademarks of Digital Equipment Corporation. Silvar-Lisco is a trademark of Silvar-Lisco. Tektronix is a trademark of Tektronix, Inc.

President Roy L. Russo

FROM THE PRESIDENT

Board acts to cut expenses, increase revenues

The Board of Governors has approved increases in the subscription prices of all Computer Society periodicals. These increases, effective in 1987, were necessary to meet rising production and distribution costs and to curb the deficits generated by some of the magazines in recent years.

In addition, the board approved a $5.00 increase in the society membership fee for 1987. This increase, which will bring the society basic membership fee to $15.00, is the first increase in member dues since 1983. The increase was approved in order to help the Computer Society maintain a stable financial position despite the current downturn in the computer industry and the attendant reduction in many society programs and revenues. Among the difficulties with which the society's leaders have had to cope are a radical reduction in the funds heretofore generated by the National Computer Conference, shortfalls in advertising revenues from our magazines, reduced attendance at many society-sponsored conferences and tutorials, and reduced income from publication sales. Even with the increase, member dues will still represent well under 10 percent of total society revenues, an exceptionally low proportion compared with similar professional societies.

The board also took many actions directed at reducing expenses, eliminating well over a half-million dollars in expenditures in 1986. Taken together, the board's actions to constrain expenses while cautiously increasing fees and income constitute a prudent plan to assure that the Computer Society will weather the current storm—maintaining its financial stability during the computer industry downturn, yet remaining positioned to resume its program of expanded services to its membership and the profession when the industry recovers.

Best regards,

Roy L. Russo

Roy L. Russo

Tom Cain now acting-president

Shortly after sending the above message, Computer Society President Roy Russo was hospitalized for heart surgery. As of press time, the news was encouraging: the surgery was successful, and Roy is experiencing a good recovery. The arrangements for the continuing operation of the society's business are best explained in Roy's own words. We share with you (below) his August 22, 1986, notice to the Board of Governors, Executive Committee, and senior staff. —*Ed.*

I write to notify you that I will be undergoing heart surgery and expect to be incapacitated for a period of about three months. Therefore, as provided in the constitution of the Computer Society, effective August 23, 1986, First Vice President Tom Cain will be serving as acting-president. Tom will remain acting-president until I issue a notice revoking this directive. In the very unlikely event that I remain incapacitated into 1987, the then first vice president will become acting-president.

Tom and I have worked closely together, and I can assure you that the society will be in the best of hands. I am sure you will support him as much as you have me.

I will be in touch with both Tom and Michael [Elliott, executive director,] to the extent possible.

I had secretly hoped for a short break from the demands of the presidency, but this is not exactly what I had in mind!

Acting-President Tom Cain is an associate professor of electrical engineering at the University of Pittsburgh. He had been serving as first vice president for publications.

Computer Professionals

BDM wants to talk with you if you're looking for a fast-paced, innovative company. We invite you to use your academic background and entrepreneurial spirit to solve the problems that our clients bring to us. We are committed to strengthening the nation in meaningful ways—in defense, communications, space, and many other areas. We take our capabilities and services directly to our clients. By using state-of-the-art technologies, we are able to find new, more effective solutions to difficult problems.

A broad spectrum of technologically challenging opportunities await you at our 36 nationwide locations. We offer you your choice of locations including:

McLean, VA
Albuquerque, NM
Los Angeles, CA
Huntsville, AL
Dayton, OH
Columbia, MD
Seattle, WA
Tacoma, WA
Austin, TX

Positions are available in the following areas:

AI ENGINEERS
- Computing Technology
  - —Parallel computer structures
  - —Distributed computing/operating systems
  - —Parallel and multiprocessor computer systems, architectures, and distributed operating systems
- Expert Systems Technology
  - —Symbolics 3670 or comparable AI workstations
  - —Expert system development tools—KEE, OPS, S.1, Knowledgecraft, ART
  - —Multiprocessor implementations
- Optical Computing Technology
  - —Optical computing and neural analogue systems
  - —Algotecture analysis

SOFTWARE ENGINEERS
- Real-Time Software
- LANs
- TCP/IP, Ethernet, DECNET
- System Test and Integration
- Configuration Management
- Quality Assurance
- Computer Security
- Computer Integrated Manufacturing
- Ada, Pascal, FORTRAN, LISP, and SMALLTALK
- Systems Architecture
- UNIX and "C"
- Optical Disk Storage Technology

DATA BASE MANAGEMENT SYSTEMS LEADERS
- Micro-Based Data Management Systems
- IDMS/R
- VAX/VMS/MVS/UNIX
- Large Data Base Management Systems
- DATACOM/DB
- Yourdon/DeMarco Structured Methodologies
- ORACLE
- Model 204
- Relational Data Base Management Systems
- ARTEMIS

These are prime opportunities for Senior Engineers, Computer Scientists, and Information Systems professionals, preferably with an MS or PhD and 3 or more years of experience. Please send your resume with location preference to:

The BDM Corporation
Holly Marcario
Dept. CM1086
7915 Jones Branch Dr.
McLean, VA 22102

BDM

An Equal Opportunity Employer. U.S. citizenship is required. Subsidiary of BDM International, Inc.

Guest Editor's Introduction

GaAs Microprocessor Technology

GaAs will never totally replace silicon; it will be used in selected aerospace, defense, and supercomputing applications. The extent of its usage is open to question.

Veljko Milutinović, Purdue University

Gallium arsenide (GaAs) technology has finally, during its third "reincarnation," reached the VLSI level of complexity.

Actually, early work on GaAs ICs started at about the same time as early work on silicon ICs. A number of technology-related problems, however, soon caused researchers to give up on GaAs, and silicon became dominant. Some important advantages of GaAs over silicon motivated the research community to try to turn back to GaAs several times during the past two decades. These "comebacks" did not succeed until the early 1980's.

A comparison of GaAs and silicon technologies

Although a number of different GaAs IC families exist, it is possible to draw some general conclusions about the advantages of GaAs technology over silicon technology:

- Currently, for the same power consumption, GaAs is about half an order of magnitude faster than emitter-coupled logic (ECL), the fastest silicon family. (This statement applies to integration levels and fabrication processes that can be depended on for production purposes. Data from laboratory experiments can be misleading if based on conditions that are not applicable to a reliable production environment.) Many researchers predict that the speed advantage of GaAs will increase over the coming years, although some researchers are skeptical about this.
- GaAs is more radiation-hard than silicon. At this time, however, the difference is difficult to quantify. On one hand, there is an indication of huge differences (several orders of magnitude). On the other hand, some recent measurements from an experimental microprocessor design show barely any difference with respect to radiation hardness.[1]
- GaAs is tolerant of temperature variations. Its operating range is from about -200°C to about +200°C. With respect to capability for accommodating a wide temperature range, the advantage of GaAs is definite.
- Also, GaAs is better suited to the efficient integration of electronic and optic

0018-9162/86/1000-0010$01.00 © 1986 IEEE

components. The usefulness of GaAs for this type of integration is still under investigation, and if developed to the appropriate level, it may have a major impact on the area of system design in general.

Basic disadvantages of GaAs technology, as compared with silicon technology, are as follows:

- GaAs wafers presently exhibit a larger "density of dislocations" (that is, a larger number of irregularities per unit of area). Consequently, GaAs chips (1) have to be smaller in VLSI area, (2) have a smaller transistor count, and (3) have a worse yield. Of course, VLSI area can be traded off against production yield to some extent. Fabrication projections for 1987[2-4] put the transistor count for GaAs ICs below 30K. However, the *absolute* limits are higher. Recently, Japanese researchers announced a GaAs memory chip with 102,300 transistors.[5] It is believed that this integration level was achieved by serious sacrifices in the yield, and under special laboratory conditions.

In conclusion, the current integration level of GaAs is about one order of magnitude below that of silicon, and is expected to stay that way, although both technologies will improve.

- At the present time, GaAs substrates are about two orders of magnitude more expensive than silicon substrates. GaAs is intolerant of aluminum, and gold interconnects have to be used. Moreover, GaAs is brittle; wafers can be damaged easily during the fabrication of integrated circuits. However, many problems related to materials are considered temporary in nature, and one prediction states that the steady-state cost will be about one order of magnitude greater for GaAs than for silicon.[6]
- The noise margin at present is not as good as in silicon. It is often necessary to trade off chip area for higher reliability. Some GaAs IC families also exhibit a sensitivity to increased fan-in and fan-out. These problems are also considered temporary.
- Companies are currently reporting problems with the testing of designs for high-speed GaAs ICs. Testing facilities that are fast enough are simply not yet available. This situation is expected to improve.

Table 1 summarizes the most important performance-related facts. These facts reveal more about possible applications of GaAs technology. GaAs will never totally replace silicon, but will be used in selected aerospace, defense, and supercomputing applications. The extent of its usage is open to question; the "competition" between GaAs and silicon is an on-going one.

Considerations for designing GaAs microprocessors

Now that GaAs has reached the VLSI level of complexity, how should one design a GaAs microprocessor? One company that was involved in the design of both a 32-bit microprocessor for GaAs technology (namely a direct-coupled FET logic enhancement/depletion-mode metal semiconductor field-effect transistor, or DCFL E/D-MESFET) and a 32-bit microprocessor for silicon technology (in this case, complementary metal oxide semiconductor/silicon on sapphire, or CMOS/SOS) released the following data:

> The technology speed ratio (i.e., the gate speed ratio) of the GaAs and silicon devices was about 6:1. However, the ratio of processor speeds (for compiled HLL code) was only about 3:1 in favor of GaAs. The situation was even worse in the area of coprocessor design. For the arithmetic coprocessor, the speed ratio was only 2:1 in favor of GaAs.[1]

The conclusion is clear—something has to be done in the area of microprocessor architecture and microprocessor design. This special issue explores the problem, which is difficult and represents a real challenge for the research community.

Is GaAs a "different animal" to work with than silicon? To answer this question, it is necessary to identify the basic technology differences of relevance for microprocessor architecture and design. GaAs is characterized by

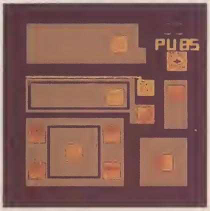
- small on-chip VLSI area and transistor count, as well as
- high ratio of off-chip to on-chip delays.

Secondary and—one hopes—temporary differences that are limited to cer-

GaAs DRAM cell research is underway at Purdue

Future GaAs VLSI architectures will require large quantities of fast, high-density memory. To date, the only memory elements available to GaAs system designers have been six-transistor static RAMs (SRAMs), which require considerable chip area and continuously consume power.

However, researchers at Purdue University, under NSF sponsorship, are developing one-transistor dynamic random access memory (DRAM) cells for use in GaAs ICs. Like the test DRAM cell shown at right, they are designed with AlGaAs/GaAs heterojunction field-effect transistor technology (variously known as MODFET, HEMT, or HIGFET technology).



A test DRAM cell developed for use in GaAs ICs. (Test cell supplied by Purdue University. Photo taken by Melgar Commercial Photographers.)

DRAMs made with this technology store electrons or holes in localized potential wells at the AlGaAs/GaAs heterojunction interface. Since the stored carriers tend to escape because of thermionic emission, the stored data is volatile (or "dynamic") and requires periodic refresh, just as silicon DRAMs do. However, since system cycle times in the AlGaAs/GaAs heterojunction field-effect transistor technology will be measured in nanoseconds, DRAMs in this technology will be feasible if storage times of a few hundred milliseconds or longer can be achieved.

Recent results obtained by J. A. Cooper, M. R. Melloch, and Q.-D. Qian at Purdue with cells similar to the one shown have demonstrated storage times of 75 seconds at 175°K, 500 minutes at 140°K, and 220 hours at 77°K (the temperature of liquid nitrogen). These storage times are sufficient for DRAM operation at cryogenic temperatures, where MODFET circuits perform best.

The Office of Naval Research is also supporting work at Purdue on DRAM cells that operate efficiently at room temperature (300°K) and above. This work may soon fill a very important gap in GaAs IC technology.

James A. Cooper
Purdue University

Table 1. Performance comparisons of GaAs and silicon ICs.

	GaAs[*1] (1μDCFL E/D-MESFET)[*4]	Silicon[*2] (2μECL)[*4]	Silicon[*3] (1.25μCMOS)[*4]	Silicon[*2] (2μCMOS)[*4]	Silicon[*2] (2μNMOS)[*4]
Complexity					
Transistor count per chip	20K to 30K	30K to 45K (TR or R)[*4]	300K	175K	200K to 300K
Chip area	Yield- and power-dependent	Yield- and power-dependent	Yield- and power-dependent	Yield- and power-dependent	Yield- and power-dependent
Speed					
Gate delay (for minimum values of fan-in and fan-out	50 ps to 150 ps	150 ps to 250 ps	500 ps to 750 ps	800 ps to 1 ns	1 ns to 3 ns
On-chip memory access (32 by 32)	0.5 ns to 2 ns	2 ns to 3 ns	5 ns to 10 ns	10 ns to 20 ns	20 ns to 40 ns
Off-chip/on-package memory access (256 by 32)	4 ns to 6 ns	6 ns to 10 ns	20 ns to 30 ns	30 ns to 40 ns	40 ns to 60 ns
Off-chip/off-package memory access	10 ns to 40 ns	20 ns to 60 ns	40 ns to 80 ns	60 ns to 100 ns	100 ns to 200 ns
IC Design					
Transistors per gate	1+(fan-in)	3+(fan-in)	2*(fan-in)	2*(fan-in)	1+(fan-in)
Transistors per memory cell					
Static	6	6	5-6	5-6	6
Dynamic	1	N/A	N/A	N/A	1
Fan-in (typical transistor size)	2 to 3 (NOR)	8	5	5	5
Fan-out (typical transistor size)	3 to 4 (NOR)	10 to 20	5	5	5
Gate delay increase for each additional fan-out (relative to gate delay with fan-out = 1)	25% to 45%	5%	20% to 30%	20% to 30%	20% to 30%

[*1] Information supplied by TriQuint.

[*2] Information supplied by Fairchild.

[*3] Information supplied by Toshiba.

[*4] Abbreviations used in this table: "DCFL E/D-MESFET," for direct-coupled FET logic enhancement/depletion-mode metal semiconductor field-effect transistor; "ECL," for emitter-coupled logic; "CMOS," for complementary metal-oxide semiconductor; "NMOS," for N-type metal-oxide semiconductor; "TR," for transistor; "R," for resistor.

tain GaAs IC families should also be taken into consideration:

- higher dependency of gate delay on fan-in and fan-out, and
- nonexistence of some register-transfer-level and lower level components, such as the pass transistor, tri-state buffer, and so on.

Small on-chip "real estate" does not necessarily impact architecture and design adversely. The impact is strong only if penalties for going off-chip are relatively high, which is the case in GaAs. Therefore, a careful partitioning of system functions across chip boundaries is extremely important. The single-chip reduced instruction set computer (RISC) type of design is probably the only viable alternative among possible approaches to GaAs CPU design.[7] For a silicon environment, one might argue for a complex instruction set computer (CISC) approach or a RISC approach; in a GaAs environment there is no such choice.

RISC design strategy, however, has to be considerably modified for GaAs. First, in GaAs there is no place for an on-chip cache memory. If memory pipelining is chosen, it is much deeper than in silicon, and code optimization algorithms developed in a silicon environment for overcoming timing and sequencing hazards in pipelined architectures are of little use in a GaAs environment. Some researchers, moreover, advocate an approach in which traditional hardware functions (like various types of interlocks, bypass buses, cache control mechanisms, and so on) are migrated into software. This approach is contrary to the silicon design environment trend during the past decade. According to this so-called "synergism methodology,[2]" the "real estate" released by hardware-to-software migration should be invested in architecture constructs (like large register files) that have been proven efficient in technologies with large penalties for going off-chip.

Finally, differences related to fan-in/fan-out and nonexistence of certain register-transfer-level (RTL) and related elements can impact low-level design solutions.

In this issue

Important research in the area of GaAs microprocessor design and architecture went into DARPA's recent effort to build a 32-bit GaAs microprocessor.[8] Most of the contributions in this special issue of *Computer* were generated as part of that program.

The article by Sherman Karp and Sven Roosild of DARPA was invited specifically for this special issue to acknowledge the important role of DARPA during the past two decades in the areas of GaAs technology and GaAs microprocessor design. The focus of the article is on the activities that preceded and enabled the realization of the old dream, a 32-bit microprocessor in GaAs technology.

Lawrence Larson, Joseph Jensen, and Paul Greiling, all of Hughes Research Laboratories, present an overview of

GaAs IC technology. Their article was included to provide the background necessary for an understanding of the field in general, and the articles that follow theirs in particular.

In their article, Barry Gilbert, Barbara Naused, Daniel Schwab, and Rick Thompson summarize the results of DARPA-sponsored efforts in GaAs system design. The emphasis is on signal-processing architectures and their implementations.

The article by myself, Alex Silbey, David Fura, Kevin Keirn, and Mark Bettinger of Purdue University, and Walter Helbig, William Heagerty, Richard Zieger, Bob Schellack, and Walter Curtice of RCA Corporation, discusses various design issues of interest for GaAs microprocessor systems. Its focus is on RCA's effort to develop an 8-bit CPU and a 32-bit CPU in DCFL E/D-MESFET GaAs technology.

Terrence Rasset, Roger Niederland, John Lane, and William Geideman of McDonnell Douglas Astronautics Company describe an effort to build a 32-bit CPU with enhancement-mode JFET GaAs technology. Numerous design decisions are discussed and explained.

In their article, Eric Fox, Kenneth Kiefer, Robert Vangen, and Shaun Whalen of Control Data Corporation (CDC) describe the design and architecture of a 32-bit CPU to be produced jointly by Texas Instruments and CDC. This CPU is based on bipolar GaAs technology.

This special issue, I believe, will have an impact on the further development of the field. Unfortunately, because of space limitations, some excellent contributions could not be included. They are scheduled to appear in future issues of *Computer.* Below, I briefly address their contents:

Douglas Fouts, John Johnson, Steven Butner, and Stephen Long of the University of California at Santa Barbara describe the system architecture of a GaAs 1 GHz tester for digital ICs.

The article by J. McDonald, H. Greub, R. Steinvorth, and A. Bergendahl of Rensselaer Polytechnic Institute concentrates on interconnection issues in wafer-scale integration for GaAs technology. (In this article, the authors propose a RISC-type architecture that is well suited for the GaAs design environment.)

In their article, Theodore Lehr and Robert Wedig of Carnegie Mellon University describe a RISC-type architecture that is oriented to the production-system environment and is well suited for implementation in GaAs technology.

Michael Morgan of Magnavox Corporation describes an architecture that is oriented to the Lisp language and is well suited for implementation in GaAs technology. The architecture is referred to as GAELIC (Gallium Arsenide Experimental Lisp IC).

Further information

Obviously, there are a number of questions that have yet to be answered. Among important sources of new ideas and results, the reader is referred to conference proceedings from the IEEE GaAs IC Symposia and the IEEE/ACM Hawaii International Conferences on System Sciences (information about these proceedings is available from the IEEE Service Center in Piscataway, N.J.) and to proceedings from the IEEE ICCD Conferences (information about these proceedings is available from the IEEE-CS West Coast Office in Los Alamitos, Calif.).

Other important sources are the specialized journals, such as *IEEE Transactions on Computers, IEEE Micro,* and *IEEE Design & Test of Computers.* ☐

Acknowledgments

This special issue could not have been realized without the help that came from various sources.

Much of the empirical research on GaAs architecture was based on an Endot Corporation software package, N.2, and the related functional description language, ISP′; these were recently donated to several universities. Much of the research on the VLSI design level was based on in-house software packages, such as RCA's CADDAS with the related hardware description language (HDL); the basic parts of the CADDAS package were recently donated to some universities. Both donations (N.2 and CADDAS) enabled the university environment to catch up quickly in the new field of GaAs microprocessor design. Therefore, new contributions from both industrial and academic environments can be expected soon.

The enthusiasm and prompt response of the reviewers deserves a very special acknowledgment. Much of the technical quality of this special issue can be attributed to their dedicated work. Their names will be included in a list of all reviewers for *Computer* magazine that is scheduled for publication in an upcoming issue.

On the IEEE-CS side, *Computer* Editor-in-Chief Michael Mulder and *Computer* Editorial Board member Wing Toy assisted in improving the final form of this special issue.

Finally, colleagues of mine at Purdue University, Robert Pierret and Mark and Sarah Yoder, provided invaluable comments and suggestions for my Guest Editor's Introduction.

References

1. W. Helbig, "RISC vs. CISC, GaAs vs. Silicon, and Hardware vs. Software," invited lecture, Joint Chapter Meeting of the IEEE Microwave Theory and Techniques/IEEE Electron Devices societies, Princeton, New Jersey, Jan. 1986.
2. V. Milutinović et al., "Design Issues in GaAs Microcomputer Systems," *Computer,* Vol. 19, No. 10, Oct. 1986 (this issue).
3. T. L. Rasset et al., "A 32-bit RISC Implemented in Enhancement-Mode JFET GaAs," *Computer,* Vol. 19, No. 10, Oct. 1986 (this issue).
4. E. R. Fox et al., "Reduced Instruction Set Architecture for a GaAs Microprocessor System," *Computer,* Vol. 19, No. 10, Oct. 1986 (this issue).
5. Y. Ishii et al., "Processing Technologies for GaAs Memory LSIs," *Proc. GaAs IC Symposium,* Oct. 1984, pp. 121-124.
6. M. R. Namordi, "Advances in GaAs Technology," invited lecture, Purdue University, West Lafayette, Indiana, Oct. 1984.
7. V. Milutinović, D. Fura, and W. Helbig, "An Introduction to GaAs Microprocessor Architecture for VLSI," *Computer,* Vol. 19, No. 3, Mar. 1986, pp. 30-42.
8. C. Barney, "DARPA Eyes 100-MIPS GaAs Chip for Star Wars," *Electronics Week,* Vol. 58, No. 20, May 20, 1985, pp. 22-23.

Veljko Milutinović is on the faculty of the School of Electrical Engineering, Purdue University. He has published over 60 technical papers and two original books and has edited four books. He is the editor of the IEEE Press *Tutorial on Advanced Microprocessors and High-Level Language Computer Architecture* and the coeditor of the IEEE Press Tutorial on *Advanced Computer Architecture.* His current interests include VLSI computer architecture for GaAs, high-level language computer architecture, and artificial intelligence computer architecture. He has consulted for Fairchild, Intel, Honeywell, NASA, RCA, and other companies. He was involved in the industrial implementation of a 32-bit VLSI microprocessor in GaAs technology, with responsibilities in the microarchitecture domain.

Milutinović received the PhD degree from the University of Belgrade, Belgrade, Yugoslavia.

Readers may direct questions concerning this special issue to Veljko Milutinović, Purdue University, School of Electrical Engineering, West Lafayette, IN 47907.

OUR MOST IMPORTANT POSITION IS OPEN

We've left our most important position open…the positioning of our company's products against today's proprietary computer systems. It takes an open mind to see past the clutter of "here today, gone tomorrow" technology, and that independent vision has led us to develop the Sun workstations. The Sun Workstation® computer system has opened up the future of high performance, technical workstations with features that include an extended UNIX* operating system, flexible design tools and the improved user interface to name a few. And while the success of our technical workstations have been acclaimed throughout the industry, we continue to search for open-minded professionals who do not believe that the development stage of computer technology is over.

SOFTWARE PRODUCTS DIVISION

The Software Products Division creates, markets and delivers quality software products that run on Sun and non-Sun computers. This means products that are focused in areas which leverage Sun's overall business and are targeted at customers who can improve their productivity with sophisticated and powerful software tools. These products will establish Sun as leader in the Computer-Assisted Software Engineering (CASE) and Artificial Intelligence (AI) markets. This also means providing a development environment that is second to none.

COMPUTER-ASSISTED SOFTWARE ENGINEERING (CASE) GROUP

This group is involved in creating a programming environment in which large numbers of software engineers can cooperate while producing complex programs.

Senior Software Engineers

Be a major player on Sun's CASE product development team. You will play a key role in determining initial product definition and in driving the project to completion. Candidates should have extensive experience in completing large development projects and be particularly knowledgeable of networking, relational databases, user interfaces and operating systems. A Bachelor's degree (Master's or PhD preferred) or equivalent professional experience is necessary, in addition to 5+ years' related experience. Mail Stop HZ

Manager of CASE Tools

Lead another new CASE project. Be responsible for acquiring, integrating and supporting computer-assisted software engineering CASE tools into the Sun environment. Requires BSCS and 6 years' experience (MSCS preferred). Mail Stop IA

PLATFORM PRODUCTS GROUP

This group provides highly technical consulting services for Sun products, particularly in support of our industry standard Network File System (NFS), as well as future products. (Prior experience as a consultant preferred.)

sun
microsystems

Open Systems For Open Minds

NFS Consultant

Work as a multi-vendor integration consultant porting NFS to UNIX and other environments, teaching NFS seminars and classes, and promoting NFS standards work. Requires a BSCS, 3–6 years' experience in UNIX kernal, distributed file systems, or networking implementations. Prefer knowledge of other operating systems (MS-DOS, VMS) and of other architectures (Intel 80386). Mail Stop IB

Window Systems Consultant

You will port a new distributed window system to many different hardware and software environments. Requires BSCS and experience in window systems implementation, UNIX internals and networking. Mail Stop IC

NFS Support Engineer

As part of an NFS "SWAT" team, you will fix bugs, write reports, do release engineering and provide backline support of end-user support organizations. Requires BSCS and 2 years' experience, UNIX and LAN experience preferred. Mail Stop ID

PROGRAMMING ENVIRONMENTS GROUP

This group builds and enhances innovative window systems and user interfaces.

Sr. Software Engineer

Take a problem area and implement a solution. Know window servers, multiprocess kernal-based window systems or single process window systems. Also be experienced with user interfaces, preferably with graphical user interfaces. Advanced degrees and real-world experience are vital, along with UNIX/C knowledge. Mail Stop IE

Software Engineer

Work in a fascinating area with some of the leaders in the field. Be able to adapt large bodies of code to changing requirements, fix bugs and maintain quality. 2-years' experience with UNIX/C required; user interface experience is preferred. Mail Stop IF

COMMUNICATIONS GROUP

This group develops data communications software for internetworking and multivendor compatibility: SNA, ISO, X.25 and XNS.

Sr. Software Engineer

You will develop data communications and networking systems solutions in a multi-vendor environment for Sun product family. Requires a background in the following areas: serial communication hardware (SIO, SCC chips) drivers, SDLC, 3270 SNA/BSC, RJE SNA/BSC, X.25, Ethernet, DECNET, TCP/IP, XNS or asynchronous protocol. BSCS required; MSCS and UNIX/C experience preferred. Mail stop IG

PROGRAMMING LANGUAGES

This group is involved with incremental compilers, advanced debugging techniques, and code optimization techniques that support advanced technologies.

Vector Processing Software Architect

In this senior role, you will construct high-performance vector processing software for workstation applications. You will be responsible for designing and implementing language extensions, and coordinating systems architecture with Hardware Engineering. Requires a minimum of 4 years' relevant experience and an MSCS; PhD preferred. Mail Stop IH

Multiprocessing Software Architect

In this senior role, you will contribute to the design and implementation of multiprocessing workstation architecture. Design and implement multiprocessing extensions to Sun programming languages. Requires 4 years of relevant experience and an MSCS; PhD preferred. Mail Stop II

Performance Analyst

You will be responsible for generating and interpreting benchmarks, in addition to acting as performance analyst for Sun's new computer architecture. You will work with languages software, particularly compilers and other generated code, to measure performance and to suggest changes that will improve performance. Applicants must have an MS in a related field (Statistics, Operations Research, Computer Science) and either a BSCS or 2 years' programming experience. Preferred candidates will have strengths in compiler and programming languages. Mail Stop IJ

Manager of Symbolic Languages (AI)

Lead the group responsible for Sun's symbolic language products and programming environmental development efforts. Play a key role in defining Sun's future symbolic language development and setting performance and quality standards. This senior position requires an extensive background in LISP and Artificial Intelligence. Advanced degree (MS) required. Mail Stop IK

Fortran Compiler

You will be responsible for writing a translator in C that accepts VMS Fortran input and produces Sun Fortran output. Requires 2 years' experience in the development of compiler front ends, experience with C and a BSCS. Knowledge of Fortran, experience with UNIX and an MSCS preferred. Mail Stop IL

Pascal Compiler

You will modify a Pascal compiler front-end. This position requires 2 years' experience writing compiler front-ends. Also, a BSCS; MSCS preferred. Experience with Pascal, UNIX/C is a plus. Mail Stop IM

Debuggers

You will create assembly-level debuggers. This position requires 2 years' experience in debugger design and 4 years' overall experience. You will also need a BSCS and knowledge of 68000 assembly language. An MSCS and 2 years' experience with UNIX/C preferred. Mail Stop IN

Shared Libraries

In this senior-level position, be responsible for Sun's approach to shared libraries as they are implemented into our most advanced technologies. You will evaluate our present efforts and generate alternative approaches, as well as generate and implement our future strategies. Applicants must be knowledgeable of operating systems (preferably UNIX internals) and compilers, and be able to interface effectively with our Open Systems Group. Requires a minimum of a BSCS and 6–8 years' solid experience, including knowledge of C. A PhD preferred. Mail Stop IO

INFORMATION MGMT GROUP

This group develops database tools that integrate relations, text graphics and data modeling advances.

Sr. Software Engineer

Be responsible for database and user interface development. Prefer candidates with strong academic credentials and experience designing graphical/mouse/menu/browsing user interfaces and designing distributed data bases. Should have knowledge of UNIX/C. Mail Stop IP

Despite the declining stability of today's high tech marketplace, Sun Microsystems, Inc., is still wide open for opportunity. Please send a resume, clearly indicating the Mail Stop code for position of interest on envelope, to: Doug Jones, Mail Stop 5-05, Sun Microsystems, Inc., 2550 Garcia Avenue, Mountain View, CA 94043. An equal opportunity employer; principals only please. *UNIX is a trademark of AT&T. The Sun Microsystems Logo is a trademark of Sun Microsystems Inc.

IEEE VIDEOCONFERENCES
SEMINARS VIA SATELLITE

DATA COMMUNICATIONS SYSTEMS

IEEE 16th Videoconference

December 4, 1986

11:00 A.M. - 4:00 P.M. Eastern Time

Data communications deals with the techniques by which machines communicate with their users, and with each other. This videoconference focuses on a few data communications issues of particularly contemporary interest: Network Security, Local Area Networks and Integrated Services Digital Networks. Each presentation will cover some of the theoretical background, progress to date, sample commercial products, and a forecast of future developments. Viewers have the opportunity to phone in questions to the presenters.

Intended Audience:
It is suitable both for current practitioners (including developers, managers and users of data communications systems), as well as those who may be considering the need to get involved with data communications. It is expected that the audience will have some experience with computers and communications, but not necessarily at a highly technical level.

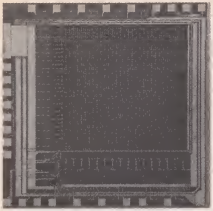
Technical Consultant and Key Presenters
Dr. Ira W. Cotton, Booz, Allen and Hamilton
Dr. Marshall D. Abrams, The MITRE Corporation
Mr. Eric L. Scace, US Sprint - Telenet Communications Corp.

The broadcast will originate live and will be available throughout the United States, Canada, Mexico and the Caribbean via satellite. An interactive network (one-way video, two-way audio) will permit telephone interaction with the presenters during the program.

Sites and Fees:
It is possible for you to have a site at your location for a license fee payable to IEEE to receive the program. For information regarding site sponsorship, please contact:

IEEE Continuing Education Department
445 Hoes Lane
Piscataway, NJ 08854-4150, USA
Telephone: (201) 981-0060 Ext. 412

IEEE



A 1K-bit GaAs complementary static RAM. (Photo courtesy McDonnell Douglas Microelectronics Center.)

DARPA, SDI, and GaAs

Sherman Karp and Sven Roosild
Defense Advanced Research Projects Agency

DARPA and the SDI support the development of GaAs technology because of its vital military applications. It has great potential for high-speed signal processing, low power dissipation, and high radiation resistance.

When silicon replaced germanium in the early 1960's as the semiconductor of choice for solid state devices, it converted the entire industry in just a few years because of two important characteristics. First, silicon has a higher energy bandgap, which permits silicon-based devices to operate over a wider temperature range (a feature especially important to the military). Second, and more important, silicon has a native oxide that provided for improved stability and planar, rather than mesa, type devices. Planar technology soon spawned integrated circuits. The integrated circuit in turn brought on the electronics revolution, allowing the complexity of circuits to increase by a factor of two every year (Moore's Law) and bringing us from single transistors to megabit memory chips.

Characteristics of GaAs

As a semiconductor, gallium arsenide has one of the characteristics that proved so important to the rapid development of silicon technology. GaAs has an even higher bandgap, allowing operation over an even wider temperature range. However, instead of a native oxide, GaAs provides a semi-insulating substrate, which neither silicon nor germanium possess. This semi-insulating substrate is ideal for an all-ion-implanted planar device technology.

Furthermore, despite the absence of a native oxide, the GaAs surface is stabilized by surface defect states (rather than passivated by a native oxide). This benefits the development of GaAs radiation-hardened circuits, of vital interest to many military systems, since radiation-induced charging of native or deposited insulators will not cause shifts in device characteristics as it does in silicon devices.

Unfortunately, the surface properties of GaAs impede commercial production of high-complexity, low-cost chips: The lack of passivation eliminates the possibility of the MOS-type circuits that form the basis for the widest segment of present day silicon VLSI.

Early efforts

In the early 1960's a lot of effort went into developing bipolar transistors in GaAs. These efforts were frustrated by the low lifetime of minority carriers in this direct-bandgap semiconductor, the difficulties of introducing dopants by diffusion, and the thermal conversion (becoming electrically conductive after heat treatments) of the semi-insulating substrates. Real progress in GaAs-based circuits began when the principal cause of thermal conversion—even slight crystal growth on the gallium-rich side of the binary phase diagram—was uncovered. Ion implantation replaced diffusion as the means for introducing dopants, and metal-semiconductor FETs (MESFETs) and junction field-effect transistors (JFETs) emerged as devices suitable for building ICs in GaAs, being majority carrier devices for which the short lifetime of GaAs is not a difficulty.

Once a workable IC fabrication technology became available, additional im-

0018-9162/86/1000-0017$01.00 © 1986 IEEE

petus to the development of a GaAs-based semiconductor technology came from the high electron mobility inherent to GaAs, which translates into improved speed and power. This advantage of GaAs plus its inherent radiation hardness made it a natural for military research. Realizing this, in the mid-1970's the Defense Sciences Office (DSO) of DARPA started a technology-based program to develop GaAs digital circuits.

At about the same time, the Strategic Technology Office (STO) in DARPA was investigating spaceborne signal processing for a broad range of military missions. STO concluded that a need existed for radiation resistant, low power, highly reliable, and programmable signal processing systems. They initiated the advanced onboard signal processor (AOSP) program to develop a suitable architecture. One of the goals of this program was the fabrication of an AOSP brassboard. After careful investigation, the best candidate technology was found to be GaAs.

GaAs chip manufacture

The first ingredient of reliable electronic systems is a reduced total chip count. Studies showed that a minimum of a 16K-bit static RAM was necessary, with a strong desire for a 64K-bit SRAM. Additionally, a large configurable gate array (6000 gates) was necessary. Considering that the most complex GaAs chip successfully fabricated at that time (1980) was a laboratory-fabricated 8-by-8 multiplier composed of 1008 gates, the task ahead looked formidable and extremely risky—just the type of technical challenge for which DARPA was created.

In 1982, DSO and STO joined forces to start development of an all-GaAs prototype of the AOSP. Both complexity and yield of GaAs ICs had to increase by orders of magnitude simultaneously. Only by phenomenally increasing yield could a sufficient number of memory and gate array chips of the desired complexity be produced to realize the GaAs AOSP prototype. Clearly, it was necessary to emulate the same learning curve improvement in yield and complexity that had propelled silicon technology to the VLSI era—and in a shorter time. Fortunately, much of the methodology and equipment used by the silicon industry transfers to GaAs circuit fabrication with minor changes. Lines processing a minimum of 100 three-inch wafers a week were contractually required, based on surveys of the silicon industry showing that such a throughput is the minimum necessary to ensure a reasonable yield of complex chips.

The first pilot line program began in 1983, the next in 1985, and a third is planned for late 1986 or early 1987. In addition to developing the chips for the AOSP, these pilot lines will provide the foundation for military and commercial uses of digital circuits in advanced computers, instrumentation, and communication systems.

GaAs specifications

In setting the specifications for components manufactured on the GaAs pilot line, it was necessary to keep in mind the primary concerns for spaceborne systems: weight and power. Consequently, not raw speed but millions of instructions per second per unit of power, or MIPS per watt, became the critical figure. Despite pressure from the contractors, DARPA maintained this commitment and set the specification of the SRAM at one microwatt per bit for static power with a 10 ns access time. Similarly, for the gate arrays the requirements were 400 μW for a D-flip-flop toggling at 50 MHz. These low power goals require operation at low threshold voltages with extremely tight control on uniformity to achieve adequate noise margins. An IC fabrication process developed for these requirements can always be relaxed to accommodate the higher drive needs of fast circuits. The reverse, however, would not necessarily be true. (Furthermore, an increase of two in frequency often requires an order of magnitude increase in power.)

Because of the multiplicity of GaAs pilot lines being developed under the auspices of DARPA contracts and the resulting legacy for military use, it was recognized that some mechanism would be required to allow defense contractors not intimately familiar with these lines to exploit their manufacturing potential. Each pilot line uses a different GaAs chip technology. Since each technology possesses different strengths, the capability to fabricate the same chip design(s) in more than one of these technologies could aid system optimization. In the commercial silicon VLSI world, in general, the design of each IC must be targeted from the outset for a particular chip technology (2-micron CMOS, 1.25-micron NMOS, ECL, etc.). To refabricate the same chip design in a different silicon chip technology, nearly the entire design process must be repeated. To minimize this problem across the several GaAs pilot lines and technologies, DARPA has sponsored the development at the Mayo Foundation of a comprehensive computer aided design package, which allows the design of technology-independent chips in a straightforward manner.

CAD of GaAs

Hosted on Digital Equipment Corporation VAX computers, this large CAD software system dramatically reduces the amount of redesign effort required to change from one GaAs technology to another. Every integrated circuit, or in fact the entire processor system, is designed initially using a generic library of gate array and standard cell functional components.

The CAD system also supports a number of technology-specific libraries of functional components. Every functional component, or macro, in every technology library has a corresponding macro component in the generic library. This matching between generic and technology-specific functional components permits an entire chip or system design to be retargeted from its initial generic form into a technology-specific form, then laid out on the specific ICs for the selected GaAs technology. The retargeting process can be fully automatic, fully manual, or a combination of both, depending upon desired chip packing density (efficiency) and desired turnaround time. Simply by selecting different technology-specific libraries, the same generic design can be retargeted for any one or more of the emerging GaAs technologies. Also, at least one silicon technology will be targeted so that functionality can be tested for minimum cost.

A GaAs microprocessor

In the spring of 1984 it was decided that, in addition to the gate array and the RAM, a custom GaAs microprocessor was needed as an engine to drive the AOSP. While it is possible to construct a microprocessor from gate arrays, performance would clearly suffer. The need for a 32-bit machine added a further complication. At the time, a 10K-bit custom chip was considered the outer limit of complexity achievable in the program's timeframe. Moreover, an interest in high performance resulted in a desire to build the microprocessor on a single chip. An exhaustive search was made of the known microprocessor architectures. The only single-chip design that met all the requirements outlined above was a DARPA-developed reduced instruction set computer (RISC) design at Stanford called Microprocessor without Interlocked Pipeline Stages (MIPS).

In the fall of 1984, three Phase I contracts were competitively awarded to TI, RCA, and McDonnell-Douglas to start the construction of a single chip, all GaAs, MIPS microprocessor with a 200-MHz clock frequency. In the fall of 1985, two of the three contractors, TI and McDonnell-Douglas, were selected to fully develop the MIPS microprocessor with a floating-point coprocessor and the necessary bus and memory interface chips.

As part of the Phase I effort, each of the contractors has undertaken to produce portions of the final design, constituting 2000 to 3000 gates. None of these parts are as yet working, but these efforts will continue into Phase II. Completion of the GaAs MIPS microprocessor is scheduled for late 1987. Current projections for the microprocessor performance are 50 ns for a floating-point multiply. To obtain these speeds, a small (1K word) cache memory with a 1 ns access time will be needed. Thus, the final design will include working chips for both fast and slow memory access, as well as chips for bus and sensor interfaces. Both contractor designs will use the same core instruction set, to which several software compilers (Pascal, Ada, and C) and translators (1750a and 68000) have already been targeted.

These two contractors will also design a vector coprocessor to support the MIPS processing system. The only requirement for this vector processor is that it run the Macro Function Assembly Language (MFAL) currently used in the development of the AOSP software. This will allow an orderly transition of the planned AOSP demonstration onto the GaAs brassboard. Since careful consideration was given to using the MFAL language initially, this is considered a plus for ease of programming. Other languages will also be accommodated.

GaAs and the SDI

The Strategic Defense Initiative was formed in 1984. DARPA's GaAs Pilot Program was shifted to SDI as a fundamental part of the radiation-resistant space signal processor program. By mutual consent, DARPA has continued to manage the GaAs program for the SDI Organization. The SDI office has in turn staunchly supported both GaAs and the AOSP. The all-GaAs AOSP system is currently scheduled for completion in 1990 as one of the major technology milestones in the SDI program. Although the specifications have not been set, it is likely that this system will operate at processing rates up to several GFlops and occupy no more than half a standard rack.

Having identified radiation-hard, low power circuitry as a unique niche for digital technology, and having identified the AOSP architecture for implementation with these circuits, we have every reason to expect that GaAs-based digital semiconductor technology will become firmly established by the end of this DARPA/SDIO program. ☐

Sherman Karp is a consultant whose interests include communications and signal processing. As a principal scientist in the Strategic Technology Office of DARPA, his contributions included the Blue/Green Satellite-to-Submarine communications system, the DOD GaAs digital technology, and the miniaturized GPS receiver and associated systems. He won the Scientist of the Year award at NOSC in 1976.

Karp received a BS in 1960 and an MS in 1962 from MIT, and a PhD in 1967 from USC, all in electrical engineering.

Sven A. Roosild is assistant director for electronic sciences in the Defense Sciences Office of DARPA. While program manager, he initiated the DARPA low power, radiation hard GaAs LSI pilot line programs. His interests include high energy ion implantation technology, radiation hardening of semiconductor devices, solid state analog signal processing components, and infrared imaging.

Roosild received a BA in engineering and applied physics from Harvard in 1960 and an MA in physics from Boston University in 1966.

Readers may write to Roosild at the Electronic Sciences Division, 1400 Wilson Blvd., Arlington, VA 22209-2308.

PL/I
FOR PROS

Your existing PL/I applications *can* run on today's hot new workstations.

Because LPI-PL/I runs under UNIX, you can move programs simply by recompiling—without re-programming. And, because it's a true high-performance compiler, LPI-PL/I makes full use of the speed of those workstations.

For more information, contact Language Processors, Inc., 400-1 Totten Pond Rd., Waltham, MA 02154 (617) 890-1155.

LPI

LPI-RPGII, LPI-COBOL, LPI-PL/I, LPI-BASIC, LPI-FORTRAN, LPI-PASCAL, LPI-C, LPI-DEBUG
LPI is a trademark of Language Processors, Inc. UNIX is a trademark of AT&T

Reader Service Number 1

CALL FOR PAPERS

The 14th International Symposium on
Computer Architecture

June 2-5, 1987
Sheraton Station Square
Pittsburgh, Pennsylvania, USA

sponsored by
Computer Society of the IEEE
Institute of Electrical and Electronics Engineers (IEEE)
Association for Computing Machinery

SYMPOSIUM COMMITTEE

General Chair
Zary Segall
Carnegie-Mellon Univ

Program Chair
Dan Siewiorek
Carnegie-Mellon Univ.

International Program Chair
Wolfgang Giloi
GMD-TUB FIRST, Germany

Finance Chair
Henry Chuang
University of Pittsburgh

Tutorials Chair
Allan Fisher
Carnegie-Mellon Univ.

Registration Chair
Rob Rutenbar
Carnegie-Mellon Univ.

Publicity and Publication Chair
Roberto Bisiani
Carnegie-Mellon Univ.

Local Arrangement Chair
Maurice Herlihy
Carnegie-Mellon Univ

Submitted papers will be accepted for evaluation until **December 1, 1986.** Five copies of the manuscript (in English, not exceeding 20 double-spaced pages) should be sent to:

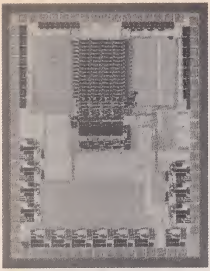
Professor D. Siewiorek
Computer Science Dept.
Carnegie-Mellon University
Pittsburgh, PA 15213, USA

Notification of acceptance will be given by **February 25, 1987.** Authors of accepted papers will be required to submit a final, camera-ready copy by **April 5, 1987.**

Papers are solicited in any aspect of Computer Architecture. Topic areas include, but are not limited to:

- Architectures for artificial intelligence applications;
- Architectures for transaction-based systems;
- Language oriented architectures;
- Operating systems oriented architectures;
- Tools and methods for architecture design and description;
- Distributed and parallel architectures;
- Performance measurement and evaluation;
- Custom architectures;
- VLSI impact on architecture.

Selected papers will be nominated by the program committee for publication in the ACM Transactions on Computer Systems.



The VE29G01 GaAs 4-bit microprocessor slice, a member of the first commercially available LSI digital logic family in high-speed GaAs. (Photo courtesy of Vitesse Electronics Corp.)

GaAs High-Speed Digital IC Technology: An Overview

Lawrence E. Larson, Joseph F. Jensen, and Paul T. Greiling

Hughes Research Laboratories

GaAs IC technology has matured to the point of fabrication of large-scale integrated circuits. It has definite potential in high-speed digital systems in the near and long term.

An earlier version of this article appeared in the *Proceedings of the International Conference on Computer Design,* October 7-10, 1985, pp. 384-390.

Gallium arsenide integrated circuit technology has advanced to the stage where small-scale integration (SSI) and medium-scale integration (MSI) circuits are available for implementation in high-speed digital systems. The recent availability of GaAs wafer foundries for fabrication of custom designs, along with commercially available GaAs components, allows system designers for the first time to take advantage of the inherent high speed and low power capabilities of the technology.

Large-scale integration (LSI) complexity circuits are already being fabricated in the United States and abroad, and higher levels of integration are expected. This will result in improved levels of performance for large digital systems. The advantages of higher levels of integration are clearly evident, although there appears to be an optimum level of integration for each GaAs logic family beyond which system speed actually degrades.

In conjunction with the development of GaAs technology, an industry-standard GaAs production process is also evolving. This generic process is available (with minor variations) from most of the GaAs wafer foundries and IC manufacturers.

Here we review digital GaAs IC device and circuit technology and analyze the performance of GaAs circuits fabricated by this production process. We also analyze the effect of the GaAs IC integration level on computer system speed.

GaAs technology: an overview

Gallium arsenide technology is attractive for the realization of high-speed digital ICs, principally because of the high electron mobility and high peak electron velocity the bulk material exhibits relative to silicon. In addition, the intrinsic carrier concentration is low enough to yield semi-insulating substrates. This property reduces device and interconnect capacitance to a low level. Other properties of GaAs, such as improved radiation tolerance and operating temperature range, can be important in some applications.

In principle, these properties can result in ICs of reduced power dissipation and increased speed, since at low electric fields (less than 10^4 V/cm) the electron velocity in GaAs is higher than that of Si. At higher electric fields, the saturated electron velocities in Si and GaAs are approximately equal, but GaAs devices require less voltage to achieve saturation.

Transistors of varying levels of process and design maturity are currently available for use in GaAs ICs. The most mature of these is the depletion-mode field effect transistor (DFET) shown in Figure 1a. The DFET, a normally-on device, has the largest current per device width of any GaAs FET. The gate structure is a Schottky diode formed on the GaAs surface. The threshold voltage of the FET (the gate voltage required to turn the transistor off)

0018-9162/86/1000-0021$01.00 © 1986 IEEE

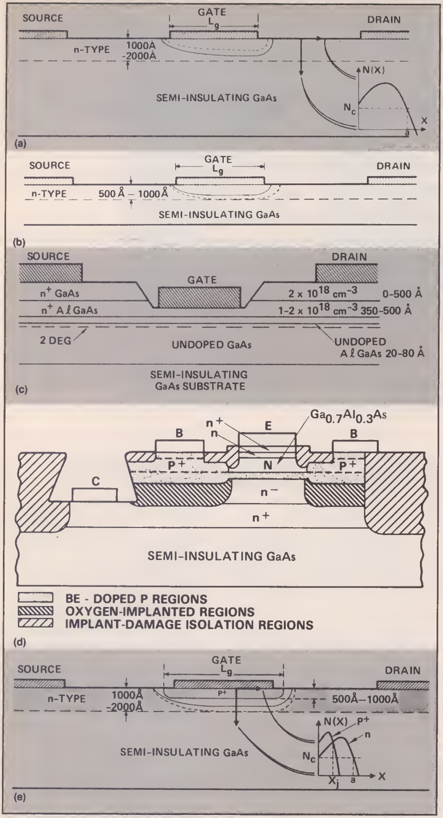


Figure 1. GaAs transistors. **(a)** shows a depletion mode field effect transistor (DFET) with a threshold voltage of −0.7V to −2.5V, a high current capacity of 0.1 mA/μm, and high power dissipation of 0.25 mW/μm. **(b)** shows an enhancement mode field effect transistor (EFET) with a threshold voltage greater than 0V, a low current capacity of 0.01 mA/μm, low power dissipation of 0.01 mW/μm, and tight process controls of less than ± 50Å. **(c)** shows a high electron mobility transistor (HEMT), which is an enhancement and depletion device with a higher gate forward bias voltage of 0.7 to 0.9V and high transconductance 450 mS/mm (300°K) to 520 mS/MM (77°K). **(d)** shows a heterojunction bipolar transistor (HBT). **(e)** shows a junction field effect transistor (JFET) with large sidewall capacitance and tight process controls. Complementary devices are available.

is determined by the channel doping and thickness. A highly doped, thick channel leads to a large negative threshold voltage.

By decreasing the channel thickness, a normally-off enhancement-mode FET (EFET) with a positive threshold voltage can be fabricated (Figure 1b). The advantage of EFET circuits is that, unlike DFET-only circuits, level shifting is not required in the logic gate. However, the voltage swing of a circuit using EFETs is limited by the forward turn-on voltage of the Schottky-barrier transistor gate (0.7V). This limits the noise immunity of the gate and places stringent fabrication requirements on threshold voltage control and uniformity. A major processing effort is currently under way in GaAs to produce EFETs with improved uniformity so that LSI circuits can be fabricated with acceptable yields.

Several recently developed GaAs devices promise improved performance over typical EFETs or DFETs. One is the high electron mobility transistor (HEMT), also known as the modulation-doped FET (MODFET) (see Figure 1c). The HEMT, or MODFET, is a heterostructure device consisting of GaAs and aluminum gallium arsenide (AlGaAs). Part of the AlGaAs layer is heavily doped to provide the mobile electrons for conduction. The GaAs layer is undoped, which reduces impurity scattering to allow maximum electron mobility. The free electrons diffuse from the heavily doped AlGaAs to the undoped GaAs, where they are confined by the energy barrier created at the heterojunction. Conduction takes place in the undoped GaAs where the electrons can travel at a high velocity. Electron mobility can be further enhanced by operating the device at low temperatures (about 77 degrees Kelvin).

Enhancement and depletion-mode devices can be fabricated in HEMT technology. HEMTs have a higher forward turn-on voltage of the Schottky-barrier gate than do MESFETs, which provides for a larger noise margin in enhancement/depletion (E/D) logic circuits. This feature is critical for obtaining higher yields at larger levels of integration. Although HEMTs are fabricated from a more complex multilayer semiconductor structure, new fabrication methods such as molecular beam epitaxy (MBE) or metal-organic chemical vapor deposition (MOCVD) may allow them to become the dominant transistor structure in GaAs LSI circuits.[1]

Another promising GaAs technology produces heterojunction bipolar transistors (HBTs) (see Figure 1d).[2] An HBT is fabricated with different semiconductor materials in its emitter and base regions. The heterostructure at the emitter-base junction allows the base to be heavily

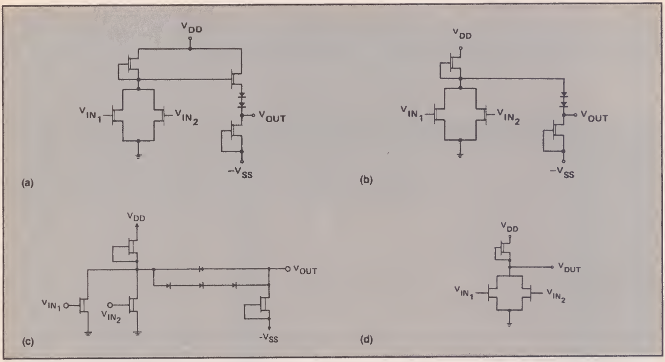


Figure 2. GaAs logic families. **(a)** shows buffered FET logic (BFL), featuring high speed with fan-out, large power dissipation (5 mW/gate), and MSI complexity. **(b)** shows unbuffered FET logic (UFL), featuring low speed with fan-out and three power dissipation (2 mW/gate). **(c)** shows capacitively enhanced logic (CEL), featuring high speed with low fan-out, low power dissipation (400 μW/gate), and LSI complexity. **(d)** shows direct-coupled FET logic (DCFL), featuring medium speed, low power dissipation (50 μW/gate), and VLSI complexity.

doped, reducing the parasitic base resistance while maintaining acceptable transistor current gain. In GaAs technology the emitter is AlGaAs, while the base and collector are GaAs. Heterojunction bipolar transistors are attractive for digital applications because of their inherently high voltage uniformity compared to a FET-based technology.

Complementary GaAs junction field effect transistor technology (JFET) has also developed impressive results[3] (see Figure 1e). In this case, the gate region is a p-n junction doide rather than a Schottky diode. The advantages of this approach include the higher turn-on voltage of the gate (0.9 to 1.2V) and the availability of both p- and n-channel devices. Although GaAs JFET technology produces devices of reduced speed (compared to MESFET technology), the availability of complementary devices and the high turn-on voltage are attractive for low-power LSI circuit implementations, such as memories.

GaAs logic families

Some of the more popular GaAs FET logic families are shown in Figure 2. The buffered-FET logic (BFL) gate (see Figure 2a) performs well with high fan-outs, but dissipates the most power (5 to 10 mW per gate).[4] The BFL gate employs only DFETs, so level shifting is required to make the input and output logic levels compatible. This extra circuitry adds both delay and extra power dissipation to the gate. However, the level shifter provides buffering to the next stage. Because of the high power dissipation, BFL is not suited for LSI complexity circuits (more than 2000 gates).[5]

A variation of BFL is unbuffered FET logic (UFL), where the level-shifting buffer is not used at the output (see Figure 2b). Instead, Schottky diodes connected at the gate output provide level shifting. UFL gates dissipate less power than BFL gates at the expense of a higher sensitivity to loading capacitance.[6]

An alternative to BFL and UFL in depletion-mode technology is the capacitively enhanced logic (CEL) gate (see Figure 2c).[7] Like UFL, CEL uses diode level shifting to establish the dc operating point of the circuit. However, a reverse biased diode, which functions as a capacitor, placed in parallel with the level-shift diodes capacitively couples the logic gates during high-speed transitions. Since the speed of the logic gate does not depend on the operating current in the level-shift diodes, the current through the level-shift diodes can be made very small, allowing very low dc power dissipation. Thus, the CEL gate is potentially faster than the UFL gate and exhibits lower power dissipation. The fan-out capability of the CEL gate is poorer than that of the BFL gate because the output is not buffered.

Note that the logic levels of BFL, UFL, and CEL can be made completely compatible, making it possible to mix the three logic families on the same chip. All of these logic families can be fabricated with the present industry standard depletion-mode GaAs technology.

Logic gates employing enhancement-mode FETs are usually designed with direct-coupled FET logic (DCFL) (see Figure 2d). These circuits have the lowest power consumption (about 50 to 300 μW/gate) of any GaAs logic family. There is no need for level shifting with DCFL gates, so power dissipation and packing density are greatly improved. The design of the logic gates resembles that of NMOS circuitry, although the gates operate at a much lower power supply voltage. DCFL circuitry is generally acknowledged as the preferred logic design strategy for realizing LSI and VLSI GaAs ICs. (Since enhancement and depletion devices are also available in HEMT technology, HEMT circuits can also be realized with any of the above logic families.)

The availability of complementary

GaAs FET technology, although still in its infancy, promises logic circuits of even lower dissipation than DCFL circuits. The poor performance of p-channel GaAs FETs precludes logic gate designs that employ an equal number of p- and n-channel devices. However, the application of techniques where only one p-channel FET is used per logic function looks very promising.

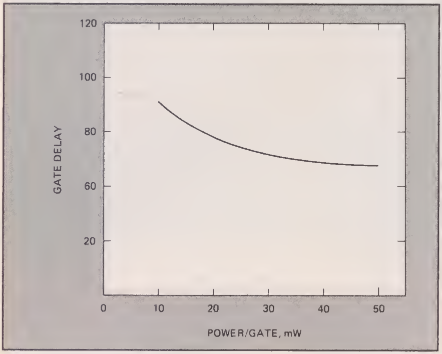


Figure 3. Speed versus power performance of BFL for 1 μm foundry technology.

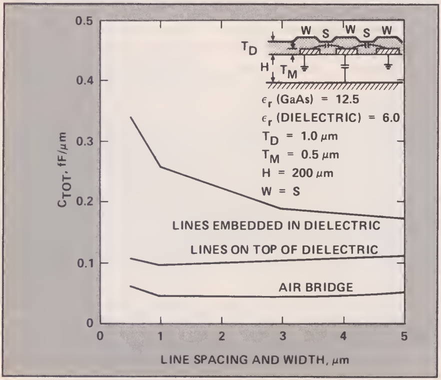


Figure 4. Interconnect capacitance calculation.

Performance of current GaAs technology

At present, GaAs ICs of LSI complexity are being fabricated primarily with DCFL logic gates. Examples of state-of-the-art circuits include 16K-bit memories with 4.1-ns access times and 16-by-16 multipliers with 10-ns multiply times. However, these circuits, still considered experimental, represent technology not expected to be commercially available for a few years yet.

At this time, digital GaAs ICs are available from a variety of sources, including Harris Microwave Semiconductor and Gigabit Logic. Wafer foundry services are available from TriQuint Semiconductor, Honeywell, Adams-Russell, Rockwell, and others. These facilities produce DFET circuitry with gate lengths of approximately one micrometer and threshold voltages of approximately −1 to −2V. Because of this commonality, we can estimate the potential performance of the circuits produced.

BFL gates typically demonstrate delays of approximately 90 ps and a sensitivity of 20 ps/fanout. A plot of the measured speed versus power performance for a BFL gate fabricated with a foundry process is given in Figure 3. The UFL gates demonstrate a typical gate delay of 150 ps, with a sensitivity of 35 ps/fan-out. Table 1 provides an estimate of the performance levels of the logic families discussed previously.

The performance of GaAs technology can be improved in a variety of ways. For example, a great deal of effort has been directed toward reducing interconnect capacitance. Figure 4 shows calculations of capacitance for interconnect lines directly on a GaAs substrate, lines on top of a dielectric layer on GaAs, and airbridge interconnect lines. Airbridge technology promises a significant reduction in interconnect capacitance. Figure 5 shows an example of airbridge interconnects for a GaAs logic gate fabricated at Hughes Research Laboratories.

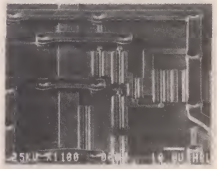


Figure 5. Airbridge interconnect technology.

Table 1. Typical logic family performance for different technologies.

Logic Family	Speed (ps)	Fan-out Sensitivity (ps/F.O.)	Capacitance Sensitivity (ps/fF)	Power (mW/Gate)
BFL (1 μm)	90	20	0.67	10.0
BFL (0.5 μm)	54	12	0.67	10.0
UFL (1 μm)	146	35	2.8	2.5
CEL (1 μm)	100	40	2.22	0.36
DCFL (1μm) (E/D)	54	35	1.84	0.25
HEMT (0.5 μm) (E/D) (77°K)	11	7	0.32	1.3

Another area where logic gate performance can improve is in submicrometer lithography. The microphotograph in Figure 6 show MESFETs with 0.6-μm, 0.4-μm, and 0.2-μm gate lengths fabricated with electron-beam lithography. Submicrometer gate length transistors demonstrate a higher transconductance and lower input capacitance, which translate into improved logic gate speed and reduced power dissipation.

Optimum integration levels for GaAs computer systems

The complexity and potential speed of GaAs ICs should improve substantially in the near future. This technological improvement will affect computer system performance in two ways. First, improved design rules will increase the intrinsic speed of the logic gates and second, increasing levels of integration will reduce the delay time associated with chip-to-chip and board-to-board communications.

A detailed analysis of these effects on the performance of a hypothetical GaAs computing system follows. The analysis does not optimize the architecture to reduce chip-to-chip and board-to-board communications and, as a result, tends to be pessimistic at small levels of integration. The other pessimistic assumption is that the GaAs technology is constant throughout the system. A realistic GaAs computer system would probably employ a mixture of E/D, bipolar, and complementary circuits, each optimized for the required task. However, the analysis does provide an acceptable indication of the trend in system speed with increasing levels of integration.

A typical large computer is composed of ICs mounted on printed circuit boards, with the circuit boards stacked on top of each other. Figure 7 is a schematic of the layout on a printed circuit board of a typical computer system.[8] This sample layout illustrates the delays caused by wiring on the chip and board as well as the intrinsic gate delay and I/O buffer delay.

The machine cycle time, τ_{mc}, for a given architecture can be approximated by the following relationship[9]:

$$\tau_{mc} = [P_i(D_i + T_i) + P_e(D_e + T_e) + P_cT_c]\ L$$

where L is the number of stages in the critical path of the machine; P_i, P_e, and P_c are the probabilities of the next gate in the critical path residing on the same IC, on another IC on the same board, or on a different board, respectively; D_i and D_e are the average gate delay of the on-chip logic gates and chip-to-chip buffers, respectively; and T_i, T_e, and T_c are the signal propagation delay on-chip, chip-to-chip, and between boards, respectively.

The power dissipation per chip is fixed at some constant maximum value, limited by the cooling capacity of the interconnect technology. The average number of stages in the critical path, L, typically lies between 10 and 15 for most high performance computer systems.[10]

A Rent's rule relationship governs the number of I/O per chip.[11] That is,

$$N_{I/O} = 0.7\ (\text{gates/chip})^{0.6}$$

In addition, the wire length, both on-chip and off-chip, is governed by a weak exponential dependence[12]:

$$\text{wire length} = K\ (\text{gates/chip})^{0.1}$$

where K is an empirical constant determined by published data on chip and board interconnections.

The gate delays D_i and D_e are determined by technological integration-level and power-dissipation factors as follows: The power dissipated in the I/O buffers is determined by the power required to drive a 1V signal into a 50Ω characteristic board impedance. The power dissipation per gate is determined from the difference between the total specified chip power and

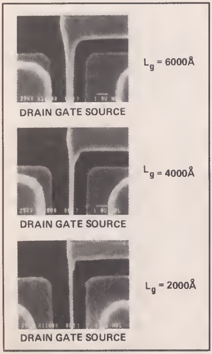


Figure 6. Submicrometer gates fabricated by E-beam lithography (EBL).

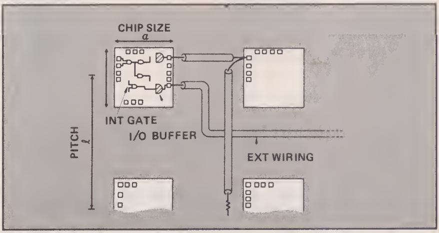


Figure 7. Sample of computer system layout.

Table 2. Logic gate delay coefficients for GaAs logic families.

Logic Family	D_0 (ps)	D_F (ps/F.O.)	D_C (ps·mW/fF)	C_p (fF)
BFL (1 μm)	37.7	20	6.7	38.2
BFL (0.5 μm)	19.5	12	6.7	19.7
UFL (1 μm)	20.0	35	7.0	32.5
CEL (1 μm)	37.7	40	0.8	10.0
DCFL (1 μm) (E/D)	10.0	35	0.46	4.4
HEMT (0.5 μm) (E/D) 77°K	2.8	7	0.43	2.7

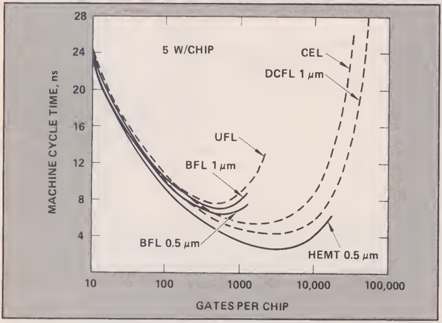


Figure 8. Machine cycle time versus chip complexity at 5W per chip for different logic families with 500,000 gates and 15 gate delays per machine cycle.

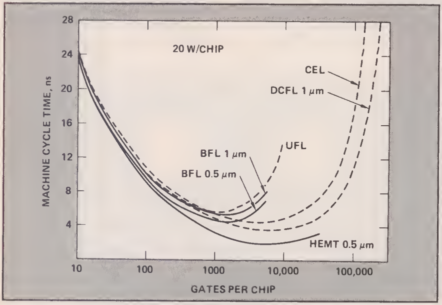


Figure 9. Machine cycle time versus chip complexity at 20W per chip with 500,000 gates and 15 gate delays per machine cycle.

the power required for I/O buffers, divided by the total number of gates. A portion of this gate power is available to drive the interconnect and fan-out capacitance. The gate delay is described by

$$D_i = D_o + [(D_f)(fo)] + (D_c/P_g)(C_f + C_w)$$

where

D_o = intrinsic gate delay (ps)
D_f = sensitivity to fan-out (ps/fo)
D_c = speed-power coefficient, which characterizes the gate sensitivity to load capacitance (ps·mW/fF)
C_f = fringing capacitance that does not scale with gate size (pF)
C_w = wire capacitance (pF)
P_g = power per gate (mW)
fo = fan-out

The coefficients D_o, D_f, and D_c and fringing capacitance, C_f, can be calculated from technological factors or can be fitted to measured data. Table 2 gives the value of these coefficients for several technologies from measured data and simulated results. The coefficients for the BFL and UFL logic families are determined from measured data. The coefficients for CEL are based on simulations that assumed a process optimized for CEL gates. The coefficients for HEMT and MESFET DCFL gates were estimated from published data.[1]

D_e is determined by calculating the minimum size device required to drive a 50Ω characteristic board impedance, which determines the capacitance of the buffer and its intrinsic speed.

The results show that computer system performance can improve through higher levels of integration and packaging that allows for higher power dissipation per IC. The system speed of each logic family improves as the level of integration increases until an optimum integration level is reached, after which system performance falls. Lithographic limitations prevent levels of integration from rising indefinitely. Performance reaches an optimum value at a lower level of integration because the I/O buffers rob power from the inner logic gates.

Figures 8 and 9 show the results of two different simulations of machine cycle times for a GaAs computer system for the technologies described earlier. It is assumed that the total system contains 5×10^5 gates with 15 delay stages in the critical path. At low levels of integration, the system speed is dominated by chip-to-chip interconnect delay. As a result, there is little difference between the speed of a computer realized in any one of these technologies (although power dissipation and volume differ).

The advantages of a lower power logic family, such as DCFL, are evident at higher levels of integration. The optimum level of integration for BFL is about 500 gates at the 5W per chip level. The optimum level of integration for both MESFET and HEMT DCFL circuits is about 3000 gates at the 5W per chip level.

Allowing higher power dissipation for each IC increases the optimum level of integration and improves the system performance. For instance, the maximum heat flux for liquid cooling of an IC is 20 W/cm^2 under optimum conditions. Assuming the IC package can be designed to take advantage of this limit and maintain acceptable reliability, a power dissipation of 20W per chip could be feasible. At this level, simulations show the optimum level of integration for DCFL increasing to approximately 6000 gates. As a result, the system cycle time for GaAs MESFET and HEMT circuits is reduced from the 5W per chip level.

These results are based on an analysis assuming a somewhat random architecture not optimized for minimization of the path length or I/O interconnect. Therefore, individual architectures can substantially exceed these performance parameters. For example, the Cray 1 achieves a cycle time of 12.5 ns with silicon technology by reducing the number of gates in the critical path to eight and by observing strict timing rules. The Cray 2 achieves a 4.1-ns cycle time with silicon ICs and advanced cooling techniques.

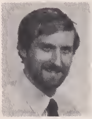
The use of GaAs technology in digital systems promises reduced power consumption and improved speed. The presently available industry standard GaAs process is based on depletion-mode technology, which limits the optimum level of integration to approximately a thousand gates. Logic circuits designed with this technology exhibit unloaded gate delays of less than 100 ps. Emerging technologies such as DCFL offer even higher speeds, lower power consumption, and higher levels of integration. Computer systems based on MESFET and HEMT DCFL should achieve optimum performance when the integration level approaches 5000 gates. □

Acknowledgments

The authors thank Ronald Lundgren and Michael Waldner for their valuable comments, and are grateful to the reviewers for their incisive suggestions on the manuscript.

References

1. H. Morkoc and P. M. Solomon, "The HEMT, a Superfast Transistor," *IEEE Spectrum*, Feb. 1984, pp. 28-35.
2. P. Asbeck et al., *Proc. 1983 IEEE GaAs IC Symp.*, Nov. 1983, pp. 170-173.
3. R. Zuleeg et al., "Femtojoule High-Speed Planar GaAs E-JFET Logic," *IEEE Trans. Electron Devices*, Vol. ED-2, June 1978, pp. 628-639.
4. R. L. Van Tuyl and C. A. Liechti, technical report AFAL-TR-74-40, Air Force Avionics Lab., contract F33615-73-C-1242, Mar. 1974
5. P. T. Greiling et al., *Lambda*, Vol. II, No. 1, first quarter 1981, pp. 44-47.
6. A. Barna and C. A. Liechti, *IEEE J. Solid-State Circuits*, Aug. 1978, pp. 708-716.
7. P. Mellor and A. W. Livingstone, "Capacitor-Coupled Logic Using GaAs Depletion-Mode FETs," *Electronics Letters*, Vol. 16, No. 19, Sept. 1980, pp. 749-750.
8. M. Abe et al., *IEEE Trans. Electron Devices*, July 1982, pp. 1088-1093.
9. E. Bloch and D. Galage, *Computer*, Apr. 1978, pp. 64-76.
10. T. Chiba et al., *IEEE Trans. on Components, Hybrids and Manufacturing Tech.*, Mar. 1981, pp. 107-113.
11. P. M. Solomon, *Proc. IEEE*, May 1982, pp. 489-504.
12. W. E. Donath, *IEEE Trans. Circuits and Systems*, Vol. CAS-26, No. 4, Apr. 1979, pp. 272-277.



Lawrence E. Larson joined Hughes Aircraft Company in 1980, where he was involved in the design and fabrication of analog and digital MOS integrated circuits. In 1984 he moved to Hughes Research Laboratories in Malibu, California, where he has been involved in research on high-speed integrated circuit design and semiconductor device modeling.

Larson received BS and ME degrees in electrical engineering from Cornell University, Ithaca, NY, in 1979 and 1980. Currently, he is a Hughes Aircraft Doctoral Fellow pursuing a PhD at the University of California, Los Angeles.

Joseph F. Jensen joined Hughes Aircraft Company in January 1981 on the fellowship rotation program. His job assignments have included work on high-speed A/D converters in silicon bipolar technology, VLSI design in CMOS/SOS technology, and VLSI testing methodology using electron-beam testing techniques. During this time he also attended UCLA on a Howard Hughes fellowship. In August 1983 Jensen joined the staff of Hughes Research Laboratories in Malibu, California, where he has played a key role in the development of high-speed GaAs circuits.

Jensen received a BS and an MS degree in electrical engineering from the University of California, Los Angeles, in 1980 and 1983.

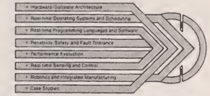
Paul T. Greiling joined the staff at Hughes Research Laboratories in Malibu, California, in 1976. He has been responsible for the design, modeling, and testing of GaAs digital ICs. At present he is Head of the GaAs IC Design and Analysis Section, working on high-speed GaAs logic circuits. He is also an Adjunct Professor in the Electrical Sciences and Engineering Department at UCLA.

Greiling received a BSEE degree and a BS degree in mathematics in 1963 from the University of Michigan, Ann Arbor, Michigan. He subsequently received an MSEE degree in 1964 and a PhD in 1970 from the same university.

Readers may write to Larson at Hughes Research Laboratories, 3011 Malibu Canyon Rd., Malibu, CA 90265.

Preliminary Program

1986 Real-Time Systems Symposium


• Hardware-Software Architecture
• Real-time Operating Systems and Scheduling
• Real-time Programming Languages and Software
• Reliability Safety and Fault Tolerance
• Performance Evaluation
• Real-time Sensing and Control
• Robotics and Integrated Manufacturing
• Case Studies

December 2-4, 1986 • Fairmont Hotel • New Orleans, LA

Sponsors:
The Computer Society of the IEEE
TC Database Systems
TC Distributed Computing Systems
TC Fault-Tolerant Computing
TC Operating Systems
TC Software Engineering

Scope: This Symposium addresses all aspects in the design, development, and analysis of real-time systems.

General Chair: John A. Stankovic, University of Massachusetts

Program Chair: Kang G. Shin, University of Michigan

Tuesday, December 2, 1986

8:00-8:45 Registration
8:45-9:00 Welcoming Ceremony
9:00-10:00 Programming Systems

- RNet A Hard Real-Time Distributed Programming System—Chantal Belzile, Michael Coulas, Genevieve Marquise, Glenn H MacEwen, Queen's University
- Software Clocks, Concurrent Programming, and Slice-Based Scheduling— Alan C Shaw, University of Washington

10:00-10:30 Coffee Break
10:30-12:00 Architecture

- An Architecture and an Interconnection Scheme for Time-Sliced Buses in Real-Time Processing—A Kovaleski, S Ratheal, Fabrizio Lombardi, University of Colorado
- Realtime Response on a Message Based Multiprocessor—Robert Olson, ELXSI
- A Real-Time Distributed Processing System— Timothy G. Saponas, Intel Corporation

12:00-13:30 Lunch
13:30-15:00 Fault Tolerance I

- Real-Time Fault-Tolerant Design for VLSI Array Processors—S Y Kung, C. W Chang, Chein-Wei Jen, University of Southern California
- On-Line Detection of Errors in Systolic Arrays—Jung-Hwan Kim, Sukhakar M. Reddy, University of Iowa
- A New Concurrent Memory-Reference Checking Technique—Hossam S Hassanein, Kenneth C Smith, University of Toronto

15:00-15:30 Coffee Break
15:30-17:00 Reliability/Performance Modeling

- A Probabilistic Model of Algorithm-Based Fault Tolerance in Array Processors for Real-Time Systems—Jacob A Abraham, Prithviraj Banerjee, University of Illinois at Urbana
- Implementation and Reliability Estimation of Dynamic Redundancy Networks—Menkae Jeng, H. J Siegel, Purdue University
- Analysis of the Execution Time of Real-Time Tasks—Michael Woodbury, The University of Michigan

Wednesday, December 3, 1986

8:00-9:00 Performance Analysis

- A Graph-Theoretic Approach for Timing Analysis in Real-Time Logic— Farnam Jahanian, Aloysius K Mok, University of Texas at Austin
- Performance Modeling and Measurement of Real-Time Multiprocessors with Time-Shared Buses—Michael Woodbury, The University of Michigan

9:00-10:00 Intercommunications

- A Virtual Time CSMA Protocol for Hard Real-Time Communications—Krithivasan Ramamritham, Wei Zhao, University of Massachusetts
- Protocols for Timed Synchronous Process Communications—S B Davidson, Insup Lee, University of Pennsylvania

10:00-10:30 Coffee Break
10:30-12:00 Operating Systems

- A Distributed Kernel Model for Reliable Group Communication—K Ravindran, Samuel T Chanson, University of British Columbia
- High-Performance Operating System Constructs for Real-Time, Robotics Software—Win Bo, Prabha Gopinath, Karsten Schwan, Ohio State University
- Real-Time Performance of a Completely Distributed Operating System—Douglas C Daniels, Horst F Wedde, Wayne State University

12:00-13:30 Lunch
13:30-15:00 Scheduling

- Dynamic Scheduling of Groups of Tasks with Precedence Constraints in Distributed Hard Real-Time Systems—John A Stankovic, Krithivasan Ramamritham, Shengchang Cheng, University of Massachusetts
- Distributed Scheduling Under Deadline Constraints A Comparison of Sender-initiative and Receiver-initiative Approaches—Hung-Yang Chang, Miron Livny, University of Wisconsin
- Solutions for Some Practical Problems in Prioritized Preemptive Scheduling —John P Lehoczky, Ragunathan Rajkumar, Lui Sha, Carnegie-Mellon University

15:00-15:30 Coffee Break
15:30-17:00 Task Allocation/Load Balancing

- An On-Line Algorithm for Real-Time Tasks Allocation—Sadegh Davari, Sudarshan K Dhall, University of Oklahoma
- A Study of Quasi-Dynamic Load Balancing in Soft Real-Time Distributed Computer Systems—Suresh Singh, James F Kurose, University of Massachusetts
- Instruction Level Mechanisms for Accurate Real-Time Task Scheduling—Richard A. Volz, Trevor N Mudge, The University of Michigan

Thursday, December 4, 1986

8:30-10:00 Fault Tolerance II

- Error Propagation in a Digital Avionic Processor A Simulation-Based Study—D Lamelino, Ravi K Iyer, University of Illinois at Urbana
- Analysis of Recovery Protocols In Distributed On-Line Transaction Processing Systems— Balakrishna R Iyer, Philip S Yu, Yann-Hang Lee, IBM T J Watson Research Center
- Practicality of Non-Interfering Checkpoints in Distributed Database Systems—Sang H Son, University of Virginia

10:00-10:30 Coffee Break
10:30-12:00 Applications

- A Real-Time Operating System for a Nuclear Power Plant Computer—Jaynarayan H Lala, Linda S Alger, Charles S Draper Lab, Inc
- A Real-Time Juggling Robot—David H Jameson, Marc D Donner, IBM T J Watson Research Center
- Real-Time Direct Kinematic on a VLSI Chip— Steven S Leung, Michael Shanblatt, Michigan State University

THE COMPUTER SOCIETY OF THE IEEE

THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, INC
IEEE

To Make Reservations at the Symposium Hotel call or write directly to:
RTSS '86
Fairmont Hotel
University Place
New Orleans, LA 70140
(504) 529-7111

For more information contact:
Professor Jack Stankovic
General Chairman
University of Massachusetts
Amherst, MA 01003
(413) 545-0720

1986 Real-Time Systems Symposium Registration Form

Name ______________________________

Company ______________________________

Address ______________________________

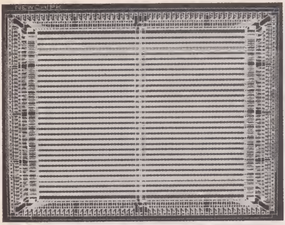
City/State/Zip ______________________________

Phone ______________________________

IEEE member # ______________________________

Mail Registrations to:
RTSS '86
Computer Society of the IEEE
1730 Massachusetts
Avenue N.W.
Washington, D.C.
20036-1903

	Advanced (prior to 11/17/86)	Late (after 11/17/86)
IEEE MEMBER	$105	$150
NON-MEMBER	$140	$190
STUDENT	$ 35	$ 55



The floorplan of the Honeywell 5183, a 5500-equivalent-gate GaAs gate array developed by Honeywell, Inc., Rockwell International, and the Mayo Foundation. (Photo courtesy of the Mayo Foundation.)

Signal Processors Based Upon GaAs ICs:

The Need for a Wholistic Design Approach

Barry K. Gilbert, Barbara A. Naused, Daniel J. Schwab, and Rick L. Thompson
Mayo Foundation

Recasting traditional signal-processor architectures at the memory-layout, logic-design, arithmetic-implementation, and system-architecture levels will permit fuller use of GaAs technology.

Advances in the development of digital GaAs ICs have progressed to the point that designers of signal and data processors and of analog conversion modules can begin to discern the systems applications for which they are best suited. The high electron mobilities of GaAs transistors result in very fast electron transit times across their active regions and hence generate the potential for extremely short gate propagation delays.[1] So-called first-generation GaAs gates using FET structures (both depletion-mode MESFETs, or D-MESFETs, and enhancement-mode junction FETs, or EJFETs) similar to those fabricated in silicon have exhibited switching delays as low as 100 ps at 1-3 mW power dissipation. First-generation GaAs components that have been manufactured at reasonable yield levels are currently in the 1000-2000 gate range, with components of 5000-6000 gate density presently in development and demonstration.[2,3]

More advanced GaAs transistor technologies are also undergoing development that rely upon enhancement- and depletion-mode GaAs FET devices working together (so-called E/D MESFET structures); these devices have demonstrated 52-ps switching delays at 0.55 mW power dissipation.[4] In addition, an even more sophisticated second generation of GaAs transistors is being studied, i.e., a heterojunction structure combining alternate layers of GaAs and GaAlAs, in which the effective electron mobility of the device can be five times greater than in the first-generation GaAs transistor. Switching delays for the resulting gates have been reported[5] as low as 5.8 ps at 77° Kelvin and 10.2 ps at 25° centigrade. These heterojunction FET structures are referred to as High Electron Mobility Transistors (HEMTs), Modulation-Doped FETs (MODFETs), or Selectively Doped Heterojunction Transistors (SDHTs). Heterojunction GaAs bipolar transistor (HBT) equivalents of silicon bipolar and silicon IIL bipolar transistors have also been developed, with both demonstrating considerable promise as well.[6,7]

As illustrated by the previous comments, the natural operational regime of GaAs digital ICs appears to be in high bandwidth, high clock rate signal and array processing, and in supporting disciplines such as wide bandwidth short burst data recording. Optimum exploitation of the high gate speeds typical of GaAs devices will require a different approach to processor architectures than for silicon VLSI. As will be emphasized later, the number of gate delays that occur on each IC should be maximized, thereby minimizing the system performance loss caused by the interchip connections. A reconsideration of architectural issues will become

0018-9162/86/1000-0029$01.00 © 1986 IEEE

even more crucial when second-generation heterojunction transistors (e.g., the HEMT structures) can be used[5] to fabricate gates with propagation delays as low as 10-50 ps. The best mechanisms for exploiting such outstanding transistor performance for logic, arithmetic, and memory must be ascertained, and the architectural designs for entire systems will have to be reconsidered as well.[8] In the following sections we will discuss these issues for logic and for arithmetic, and then for memory components; we will also comment on the manner in which these elements may be configured in large processor systems.

The remainder of this article will expand upon the problem of designing processor systems in GaAs through the discussion of an additional set of topics that, until recently, have been of particular concern, not to the computer architect, but to the computer fabricator. In the past, processor architects employing silicon have been able to neglect many important implementation and fabrication issues, in the certain knowledge that these issues would be addressed at a later time in the system evolution by individuals experienced in IC packaging, system cooling, intraprocessor communications, and power handling. One of the most important lessons to have emerged from GaAs is that these heretofore mundane issues have, with the advent of high system clock rates, become inseparable from the architectural design itself, and can represent the difference between success in implementing the processor and total project failure. These ancillary areas are now very central to processor design and quite closely interdigitated with one another, and yet are very little understood by traditional designers of signal processors. Only the adoption of a wholistic approach to the design of high clock rate signal processors will result in the type of robust high performance systems of which GaAs is capable.

Signal processing applications requiring digital GaAs

With the prior comments in mind, there is a growing consensus among many design engineers that the majority of near-term applications for digital GaAs ICs will include: high-speed acquisition and perhaps storage of very wide bandwidth pulsed, pseudo-random, or continuous stream data; real-time radar signal processing and signature analysis; electronic countermeasures; data encryption; spread spectrum communications; error detection and correction of wideband data flowing through noisy channels (thereby in effect improving the signal to noise ratio of the channel); and in communications interfaces between electronic signal processors and fiber-optic data channels. As but a single example of this large class of problems for which digital GaAs appears extremely well suited, a very serious requirement exists in the US military community to be able to characterize accurately the onset of occurrence, duration, and other characteristics of radiated electromagnetic energy, e.g., radar pulses, impinging upon a vehicle such as an aircraft. Receiving antennas and RF amplifiers mounted on the vehicle can receive the energy, increase its amplitude, and convert its frequency to lower values. If the time course of the electromagnetic pulse is to be characterized, the pulse must be sampled, or digitized, by very high-speed analog-to-digital (A/D) converters, and the digitized values stored in a small scratch pad memory, from which they can be read at more leisurely rates by a digital computer, or transmitted to the inputs of a high speed D/A converter. Such a combined A/D converter and scratch pad memory, which is referred to as a Digital RF Memory (DRFM), is described more completely in reference 9. To the present, these devices have been manufactured with silicon components, and hence have been limited to sampling rates of a few hundred megahertz and sampling resolutions of 4-6 bits. The DRFMs are of such importance that the military customers for these processors have been willing to employ silicon-based DRFMs even though their performance is considered less than optimal. Thus, there is considerable interest in the expansion of their capabilities, initially through the judicious use of GaAs and silicon components in combination, and eventually, through the design and fabrication of units based entirely upon GaAs. An example of the performance achievable with GaAs-based DRFMs will be presented later in this article.

Interdependence of algorithms, architectures, and device technologies

As an extension of these comments, it should be noted that nearly a generation of architecture research for general-purpose computers, and especially for digital signal processors, has demonstrated that there is a close relationship between the type of algorithm requiring implementation, the processor architecture with which such a design is implemented, and the transistor technology building blocks from which the architecture is constructed. Recent research regarding the optimum use of silicon VLSI has assumed the availability of a large number of gates on each IC, while downplaying the utility or even the effects of gate speed. Hence, VLSI circuit designers have been motivated to employ highly parallel signal processor architectures, and have placed less emphasis on structures that exploit gate speed. Only recently has the Very High Speed Integrated Circuits (VHSIC) Program, sponsored by the US Department of Defense, placed more emphasis on speed requirements for large silicon components.

Conversely, a different class of signal processor architectures is best suited to implementation in GaAs, in particular those designs that retain all operands on each IC for the maximum duration. Specific examples of such layouts include iterative and recursive algorithms executed on-chip at high clock rates, systolic arrays,[10] and the so-called element-per-pixel approaches as embodied in the GAP processor developed in the United States and the CLIP machines developed in the United Kingdom.[11] During the past few years, these types of iterative and array structured algorithms have received increased attention from architectural theoreticians, and are now being studied under the generic title of "cellular automata," as described in reference 12. In general, structures that are deeply pipelined and (by inference) require a minimum of decision branching appear to be the best architectures for GaAs ICs.

It is noteworthy that even when silicon VLSI can handle the majority of a signal processing task, GaAs may still be profitably employed at the "front end" of a signal processor in either of two special instances, which are often intermingled. First, the processor may be confronted with a single stream of digital data at rates of 0.1-2 gigasamples/second, usually generated by very wideband image sensors or detectors. In these cases, a single high-speed data stream must be partitioned into a set of lower rate, parallel substreams, which in turn can be further processed by silicon devices operating at much slower clock rates. The necessity for high clock rates in these "front end" units is sometimes compounded because as many as 10-20 microcycles of the processor may be required to preprocess each incoming data sample; an input data rate of 10^8 samples/second might thus demand a system clock rate as high as 2×10^9 Hz.

As noted above, the high bandwidth signal processing environment does appear to be the most natural vehicle for digital GaAs in the near to intermediate term, since signal processor architectures can often be deeply pipelined and thus can exploit high system clock rates. However, although mainframe and supercomputer architectures execute more decision

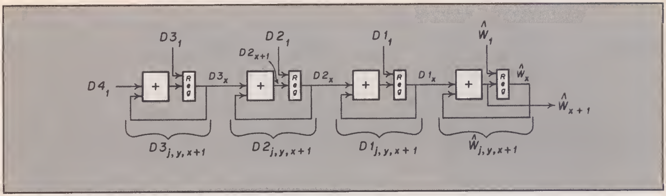


Figure 1. A fourth-degree recursion suitable for hardware implementation in GaAs digital logic; the function can be implemented either in fixed-precision or floating-point arithmetic. This implementation recirculates the partially processed data operands "on-chip" before releasing the final values, and the computational structure is very simple; both features are ideal for the gate characteristics of GaAs (see text).

branches and are usually less deeply pipelined than signal processors, it should be possible to modify traditional data processor architectures to exploit the high speeds of GaAs. It might not even be necessary or desirable to fabricate an entire mainframe with GaAs components; specific locations might be identified in the architecture, e.g., in the instruction pipes or in the cache memories of large machines, where a large benefit may be derived from the use of a relatively small number of GaAs components. It is even possible that entire supercomputer mainframes may eventually be fabricated from GaAs components. One well known US vendor of supercomputers, Cray Research, is in fact conducting a substantial GaAs research effort for its Cray III machine; apparently this vendor has been able to identify ways to modify its architecture to exploit the best features of GaAs.

The on-chip architectures employed for logic and memory components must be completely reconsidered to maximize their throughput while minimizing the amount of system performance dissipated in the intercomponent and on-chip propagation delays.[8] Simple logic, e.g., 8:1 multiplexers with two or three levels of gates, will not make efficient use of the GaAs transistor capability, particularly with the advent of the second-generation heterojunction transistors. Designers will feel compelled to maximize the number of gate stages traversed by the data operands. One effective method of increasing the number of gate stages on-chip is through iterative implementations of signal processing algorithms commonly fabricated as parallel designs. A serial implementation can often be identified by rewriting the signal processing equations in a bit-expanded form, and then operating on the least significant bits of the data operands, followed recursively or iteratively by operations on the next most significant bits, etc.; an example of this type of approach appears in reference 13).

Not infrequently it is possible to recast a problem into a completely different form, which itself can be implemented as a recursion or iteration. An example of this type of approach is the evaluation of polynomials that are being used as auxiliary equations to generate constants or parameters in a more complex system of equations. Specifically, the equation:

$$Y_i = (A_n)X_i^{\,n} + (A_{n-1})X_i^{\,n-1} + \ldots + (A_1)X_i + A_0$$

could be evaluated by performing all of the indicated multiplications and additions for all values of A_i and X_i, or else the right-hand side of the equation could be factored according to Horner's rule to decrease the number of multiplications that must be performed. However, it has been known since the eighteenth century that a polynomial of any given degree can be replaced exactly by a recursion of the same degree, in which case no multiplications need be performed at all. The recursion for a fourth degree polynomial may be written as:

$$\begin{aligned} Y(X_{i+1}) &= W(X_i) + D1(X_i) \\ D1(X_{i+1}) &= D1(X_i) + D2(X_i) \\ D2(X_{i+1}) &= D2(X_i) + D3(X_i) \\ D3(X_{i+1}) &= D3(X_i) + D4 \end{aligned}$$

where Y(1), D1(1), D2(1), and D3(1) are initial values, and D4 is a constant. Extending the approach one step further, the recursion in turn has a straightforward equivalent in hardware, as indicated in Figure 1, which illustrates the implementation for this particular recursion. A hardware embodiment of this type of algorithmic form is a natural structure for digital GaAs because of its simplicity, the relatively large number of machine cycles during which a given data operand would recirculate within the hardware, and the flowthrough nature of the resulting structure (whether implemented in fixed-precision or floating-point arithmetic). This particular method, which is described in greater detail in reference 14, has applications in synthetic aperture radar and synthetic focus sonar, and in any high-speed signal-processing application in which the evaluation of auxiliary equations must be performed to generate internal constants or parameters for a larger algorithmic process.

Of the wide variety of approaches at the disposal of the designer to tailor architectures for the features of GaAs logic gates, some, such as those described above, can be employed to implement common functions such as fixed precision multiplication, to calculate inner products or matrix products, or to conduct special operations such as recursions. As such, they may be employed as sections of full-scale signal processors but do not impact the global designs of these processors. However, some approaches affect the entire concept and architecture of a signal processor, as well as the detailed implementations of arithmetic operations. Examples of these techniques will be presented later, in conjunction with a discussion of global architectural issues.

Memory components

Both signal and data processors require large amounts of high-speed memory; for many signal-processing algorithms, e.g., in-place computation of the Fast Fourier Transform, memory speed is often the dominant element in overall system performance. Two types of memories must be developed for these processors: (1) extremely fast cycle time (1-3 ns) static random access memories (SRAMs) and read-only memories (ROMs) to support GaAs microprogram control units and high-speed caches (see, e.g., reference 15); and (2) somewhat slower (4-10 ns), larger

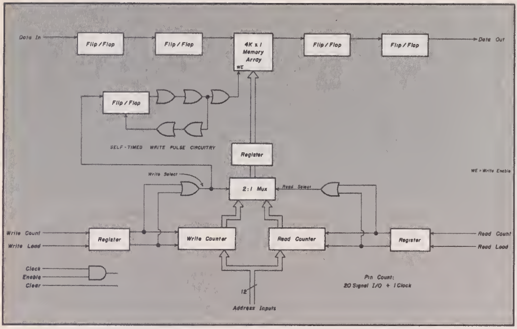


Figure 2. Pipelined implementation of a 4K × 1-bit GaAs SRAM uses self-timed write pulse circuitry and incorporates read and write counter/registers that allow presetting of initial addresses with on-chip address generation for stream read/write operations. This structure also implements interleaved read/write operations and random addressing. This architecture minimizes delivering addresses at high speed during every clock cycle; the only high-speed signal required by the memory is the clock.

memories, either SRAMs or dynamic random access memories (DRAMs) for local memory.

Although GaAs SRAMs and DRAMs exhibiting cycle times longer than approximately 5-7 ns can use conventional silicon memory component design approaches, novel configurations will be needed for memories to operate with cycle times of 4 ns or less, particularly in the peripheries of these memory chips. This situation arises because the long propagation delays typical even of small printed circuit boards make it extremely difficult to exploit conventional SRAM or ROM designs. In a conventional memory component design, address lines, chip select, and read/write signals

(1) must be applied to their lines, settle, and propagate into the memory array;

(2) must activate the memory array, and (in the case of a read operation) extract the desired bit and provide it as output data; and

(3) the retrieved data must propagate on the output lines to other portions of the processor, where it is latched into intermediate data registers.

At least 0.5-1.0 ns is required in the best case for the wavefronts to propagate to their destinations and for the input lines to settle; similar constraints exist at the outputs of the memory. A 1-ns cycle time employing "silicon memory" designs would allow less than 200 ps for each of these steps, which is unlikely to be readily achieved even in GaAs, since infeasibly short wavefront risetimes and line settling times would be required.

There is, however, a straightforward solution for this constraint. Since microprogram sequencers, arithmetic and logic units (ALUs), cache memories, and in fact entire signal processors tend to use pipelined architectures, the same type of pipelining can be introduced into the SRAM or ROM components themselves. The inclusion of output registers within the memory components would eliminate the necessity for the pipeline instruction register in a microprogram control unit, and pipeline registers in other portions of the systems as well, thereby improving performance and decreasing system complexity (that is, total chip count, though not necessarily total system gate count). An improved design for a 1- to 3-ns memory would employ on-chip pipeline registers in the address, chip-select, data-in, and data-out paths (Figure 2).

A pipelined 1-ns cycle time SRAM might require three clock cycles to fetch a given item of data: the first cycle to set up the address and chip enable lines and latch these into the on-chip pipeline registers under the control of an externally applied clock pulse, the second to fetch the data item from the memory array and place it in the on-chip output register, and the third cycle to read these values from the output register. By allowing a full 1-ns duration for each substep in the memory read or write operation, the effective distance between the memory and its source/destination hardware can be as much as 7-12 inches. Note that if this type of design approach is employed, one of the inputs to each memory chip will be a clock signal; that is, the memory will become a synchronous rather than an asynchronous component. In an additional refinement of the memory chip design, the write pulse required to write a data bit into the memory array could also be generated within the component, since the rising edge of this pulse must always occur a fixed duration after the rising edge of the clock pulse. Generation of the write pulse on the memory component itself would again decrease the number of high-speed signals transmitted on the memory board.

The foregoing description of memory access pipelining concepts still leaves a major issue unresolved. Memory components must be provided with an updated address just prior to each read or write cycle. For a memory component operating at speeds of 500 MHz or greater, the magnitude of the task of generating and distributing this stream of addresses across a page of memory is so daunting that the creation of a workable system design appears problematic at best. A small amount of additional setup time could be gained by providing two sets of address input lines, and generating consecutive addresses in different sets of hardware; an address multiplexer on the memory chip would then select in an interleaved manner between the two address input paths for the addresses supplied to the memory array.

Alternately, because the designs of most such processors are heavily pipelined, successive rather than random addresses are read from or written into their memories; instructions tend to be read in long, nonbranched sequences, while the data operands are often long vectors or large matrices with regular address sequences. Thus, an attractive substitute for off-chip address generation is the incorporation of two presettable up/down counters on each memory component, the first to control read sequences, and the second to control write sequences. Initial addresses may be preset into the counters at moderate speeds; during high-speed operation, only a clock signal and a single read/write control line need be provided to the chip at full operating rates, thereby allowing reads and writes to be interleaved at full speed (Figure 2). The memory chip write pulse could also be generated on-chip, as noted earlier.

A further enhancement of the design of the memory component would also support occasional reads and writes on random locations within memory, which could be interleaved with the stream reads and writes at full speed, though not during two or more successive memory cycles. Designs of this type have been investigated in our laboratory, and have been found to be applicable to a broad range of high-speed memory requirements. It is of considerable interest in this context that at least one merchant vendor of digital GaAs components, Gigabit Logic, plans to offer several SRAM components of various sizes and access times that do employ on-chip pipelining in the data input and data output paths, and that perform on-chip generation of the memory write pulses.[16] Further, McDonnell Douglas Microelectronics Center is fabricating an experimental, extremely low power 4K × 1-bit radiation-hardened synchronous SRAM that also includes pipeline registers on the data and address inputs, and on the data output; a 16K SRAM also under development by McDonnell Douglas is also a possible candidate for on-chip write pulse generation logic. These devices do not, however, currently support on-chip generation of the addresses for a stream read or stream write operation; it is to be expected that these additional functions will be added as the GaAs SRAM technology matures.

As a final comment regarding memory performance, it should be noted that even the eventual introduction of second-generation GaAs heterojunction transistors capable of supporting even faster memory access times will not assist the systems designer. For example, a 4K-bit × 1, 100-ps memory could not be used at its full speed, since off-chip signal lines could not conduct the signal from its source to its destination in less than 0.5-1.0 ns. However, the number of bits in each memory chip could increase markedly while retaining very fast memory cycle times in the 1- to 2-ns range. Clearly, several of these issues of speed performance versus interconnect length could be resolved by incorporating all memory bits into the same components containing, for example, the microsequencer function, since only on-chip propagation of data from the memory to its point of utilization would be required. However, there will be a continuing demand for high performance off-chip memory; it is to these requirements that the foregoing comments have been addressed.

Microprocessor architectures attractive for implementation in GaAs

In 1982, the Mayo Foundation was assigned by the Strategic Technology Office of the Defense Advanced Research Projects Agency (DARPA/STO) the task of identifying a machine architecture that could be implemented as at most a few custom ICs, but fabricated in GaAs rather than in silicon. In an attempt to minimize the development time for a microprocessor initially targeted for implementation in GaAs, silicon microprocessor architectures available at that time were reviewed for the possibility that one or more of them could be implemented in GaAs. Although a microprocessor could, of course, have been designed de novo specifically for GaAs, this approach was deemed unattractive unless plentiful software support could have been guaranteed. The possible use of an existing architecture added another constraint to the GaAs microprocessor selection effort; the design and layout of the selected microprocessor were required to be nonproprietary, readily available, and well documented.

Microprocessor designers have capitalized on the availability of large numbers of gates on VLSI chips to create relatively powerful architectures based upon structural parallelism and a rich set of assembly language instruction types; the instructions are often implemented in microcode, and the chip incorporates a microprogram sequencer and a microinstruction ROM. Because of the architectural complexity of these processors, and the desire of the designers to minimize power dissipation, the microcycle clock rates of these processors have been constrained to the range of 2-25 MHz. If high clock rate is of no consequence to the application for which these processors are intended, the use of rich instruction sets and parallel architectures is a reasonable method of achieving high computational capability.

Because GaAs digital IC fabrication technology is still relatively immature, the device densities of semicustom and custom components are currently considerably less than in silicon. As a result, one major constraint placed on the design of a microprocessor to be implemented in GaAs was that it would have to contain less than 10,000 equivalent gates, allowing its placement on a single custom chip that could be fabricated with acceptable yields by the end of the 1980's. Additional constraints included a 32-bit architecture, a 24- or even a 32-bit address bus to support at least 16M bytes of direct memory access, an efficient input/output (I/O) and interrupt handler, a high instruction execution rate, and the capability of floating-point operation, in software if not in hardware.

Several commercial or aerospace industry architectures that were examined but rejected on the basis of one or more of the criteria described above were the National Semiconductor 16032, the Texas Instruments 9900 series, the Motorola 68010, and the Intel iAPX 432. For all of these microprocessor architectures, the principal drawback was their large device counts, which were well in excess of 10,000 gates.

A microprocessor architectural approach that appeared to fit the constraints for a GaAs microprocessor more closely was represented by the family of reduced instruction set computers (RISCs). (See the article by Fox et al. in this issue for an overview of the use of RISC architecture in the GaAs microprocessors.) All of the generic RISC architectures have several features in common:

(1) a small number of simplified instructions;

(2) a fixed instruction format (all instructions are the same size and structure);

(3) hardwired rather than microcode control;

(4) single-cycle execution for most instructions; and

(5) a load/store architecture (i.e., in which only the load and store instructions access memory, while other instructions operate on registers).

The Stanford MIPS architecture was selected to serve as a baseline for the development of a 32-bit microprocessor implemented in GaAs. Three contractors, RCA, Inc., McDonnell Douglas, and Texas Instruments completed an architecture study phase in the fall of 1985, and two of these teams are presently involved in a two year project to fabricate two 32-bit GaAs microprocessors, an effort made feasible by the simplicity of the original MIPS architecture. The goal for this GaAs version of MIPS is a custom microprocessor chip of not more than 10,000 gates, a 200-MHz clock rate, and a 100 million instruction per second execution rate. A floating point coprocessor chip, also of 10,000 gate size (or less), will accompany the main processor when high-speed floating-point operations are required.

A collaborative effort among research teams at Stanford University, Carnegie Mellon University, the Mayo Foundation, Sperry Univac, Control Data Corpora-

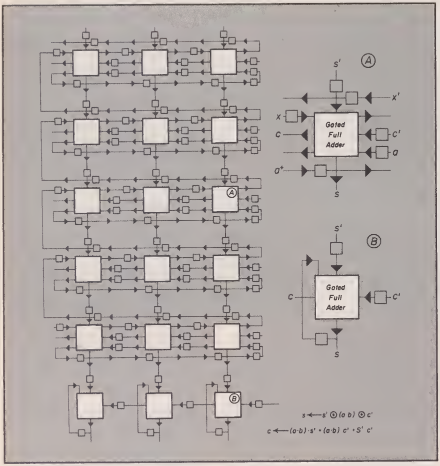


Figure 3. Implementation of a typical signal processor function, a convolver-correlator, using bit-level modular systolic arrays. Complexities of the basic functions are low (pipelined full adder is 25 gates) and all connections are single-source, single destination, and nearest neighbor; such a structure is ideal for implementation as a high-clock-rate GaAs IC. (Source: *Gallium Arsenide Technology,* 1985. Used by the courtesy of Howard W. Sams & Company, publisher.)

tion, General Electric, RCA, Inc., McDonnell Douglas, Texas Instruments, and representatives of DARPA have developed a single DOD-standard MIPS Instruction Set Architecture (ISA) document (which, with a total of 69 unique instructions, differs significantly from and improves considerably upon the 1981 version of the Stanford MIPS instruction set). The GaAs microprocessor development teams are in fact designing their processors to execute this standardized ISA. The successful creation of a standard ISA document (see reference 17) for the DOD version of the MIPS processor (now referred to as the "DOD-standard MIPS" machine) allows for flexibility at the hardware design level. In addition, the new DOD-standard MIPS ISA is currently being used as the baseline document upon which three high-level language compilers, Pascal, LISP, and ADA, are being prepared for the MIPS processors.

The original goal of the computer evaluation project conducted at the Mayo Foundation was the identification of a 32-bit microprocessor architecture that could execute at high speed, but could also be fabricated on a single GaAs chip (excluding the floating-point coprocessor) in the near future. The DOD-standard MIPS ISA meets this goal; a GaAs implementation of this new ISA should remain viable for many years, both as a stand-alone computer and as an embedded engine inside high complexity fully GaAs data and signal processors.

Architectural approaches at the system level

It was noted earlier that the special behavioral characteristics of GaAs transistors and gates must be accounted for in the design of memory, logic, and arithmetic; if designs for these functions are carried over directly from silicon VLSI, full advantage from the high speed of the GaAs transistors cannot usually be achieved. Similarly, when we wish to develop an architecture for an entire processor, we must again account for the peculiarities of GaAs transistors and gates, and the high clock and edge rates at which they function best, if a near-optimum design is to emerge. This requirement for a wholistic approach to the design of processors is not particularly necessary for implementations in silicon, but is mandatory if any of the GaAs technologies are to be employed. For example, system designs in which the data can flow continuously and uniformly between ICs that are nearest neighbors of one another, across the logic boards from one edge to the opposite edge of the same board, and between nearest-neighbor logic boards will result in minimum data path lengths and hence the fastest overall throughput. Such an approach to the layout of the ICs and their substrates will also result in minimum-complexity, lower cost logic boards and chip packages; the ability to employ only two or three signal layers rather than 10 signal layers in a multilayer printed circuit board has implications for cost control and performance improvements in these substrates. Because almost all interconnects between GaAs components behave as transmission lines and must be terminated with resistor networks that dissipate considerable power,[18] it is desirable to minimize the number of unique interconnects on a logic board or between logic boards, and to operate all such lines at the maximum possible bit rates (up to 1-2 gigabits/second/line in some cases). These constraints on chip-to-chip interconnection also have a major effect on the types of architectures that are best suited to GaAs implementation. Other equally critical issues, to be noted later, include the need to consider closely the types of intraprocessor communications structures (e.g., buses versus any of the various data exchange networks) that may be feasible to implement in GaAs chips operating at high clock rates. In the following discussion of overall global architectures that may be best suited to implementation in GaAs, the aforementioned topics must always be kept clearly in mind.

At the system architectural level, certain types of multiprocessors appear promising for implementation in GaAs because of their modularity, the amount of work performed in each node of the multiprocessor, the iterative nature of the algorithm executed in each node, or the flow-through nature of the processor architecture itself. Recently, these disparate approaches have been categorized as examples of a common architectural type referred to as "cellular automata"; these studies are attempting to develop concepts that will place such structures on a firm mathematical foundation.[12] An additional feature of several of these new architectural concepts that renders them well matched to the characteristics of GaAs components is their ability to be partitioned among numerous ICs, each of which then requires a minimum of input/output lines operating at maximum bit rates. Fortunately, the characteristics of GaAs off-chip driver transistors allow very high bit stream rates.

It may be instructive to present as an example of these architectural approaches a group of algorithms originally derived from the systolic array concept proposed by Kung and coworkers.[10] R. A. Evans and coworkers at the Royal Signals and Radar Establishment (RSRE) in the United Kingdom have generated an entire set of signal-processing algorithms for fixed-precision operand streams that execute all processing using modular bit-level systolic arrays, rather than the word-serial approaches first proposed by Kung.[10] Operations that can be performed in this bit-decomposed manner include scalar, vector-matrix, and matrix-matrix multiplication, (and hence) linear transforms, convolution/correlation (see Figure 3), and eigenvalue/eigenvector extractions.[19-20]

In these bit-decomposed algorithms, the arithmetic primitive functions are each rather small (5-200 gates), are connected to one another only by nearest neighbor links, and can be pipelined at the bit level in both the horizontal and vertical directions on a single IC. The small number of cell types (typically 1-bit full and half adders, registers, and small multiplexer/demultiplexers; see Figure 3) and their regularity of interconnection will permit a straightforward conversion of these algorithms into GaAs ICs using any of several attractive chip design strategies (see following discussion). Maximum computation will thus be performed on-chip in the environment of a moderate-sized processor "engine" operating at high clock rates, for which GaAs appears best suited. Further, these realizations appear to be extensible in vector length by cascading identical chips, while the number of high-speed off-chip signals is constrained to an absolute minimum. For example, in one silicon CMOS-based design test case conducted by Evans and coworkers, a bit-decomposed systolic array convolver/correlator for 8-bit two's-complement operands required only 13 external pins per chip, including clock, power, ground, and data lines, and up to 1024 chips could be cascaded using only nearest neighbor connections.[20]

It is of interest that the advantage of the bit-decomposed algorithm approach was originally considered by Evans and coworkers to be an improvement in the "mappability" of systolic array functions onto silicon VLSI. Initial examinations conducted at the Mayo Foundation indicate that the same benefits apply to mappings onto first- and second-generation GaAs components as well, though for quite different reasons. The RSRE group believed that the bit-level structures provided a straightforward way to cover the entire surface of a very dense VLSI component with large amounts of useful logic but with a minimum of design effort, since the same blocks could be replicated as often as desired with simple step-and-repeat operations of their chip layout software; further, a minimum of chip real estate would be devoted to interconnects. For GaAs, however, the advantages of these structures are:

- their small size and hence the ease of simulating them completely at high clock rates;
- the fact that all connections between cells are nearest neighbor and hence very short;
- that all connections are "single source, single destination," thereby decreasing the capacitive loads on the drive transistors;
- the pipelined nature of even the smallest segment of the design, thereby allowing the use of very high system clock rates; and
- their miniscule requirement for off-chip interconnects.

Intraprocessor communication problems

As the system clock rates of processors increase, a variety of physical constraints conspire to nullify many of the time-tested architectural solutions to such tasks as data communication between subsections of a large processor. One of the most troublesome of these "solutions gone awry" is the difficulty at clock rates above 50-100 MHz of employing wire-based busses to transmit instructions and intermediate data between portions of a signal processor, or to communicate with the external environment by the same mechanism. Earlier comments implied that making busses narrower, while sending the data at higher bit rates over the remaining lines, would minimize problems; nonetheless, a wire-based bus requires the connection of several drive transistors and several receiving gates to the same wire, which causes harmful transient electrical disturbances to occur on the individual bit lines of the bus. High-speed silicon ECL and GaAs components generate off-chip electrical wavefronts of 100-500 ps; these wavefronts are fast enough to contain frequency components as high as 5-10 GHz, thereby exciting the interconnects as if they were transmission lines. In turn, every source and every destination gate on a bus inevitably creates a focus for the generation of "reflections" of these fast wavefronts; the reflections in turn travel to and fro on the individual bit lines of the bus, thereby distorting and destroying the square shape of the original digital data pulses and degrading the AC noise margins of the signals.[18]

Although methods of damping these reflections are available, none is perfect and all become less than satisfactory as the data rates on the bus line rise above 100 megabits/second. At bit rates greater than a few hundred megahertz, alternative approaches must be identified. One method investigated in this laboratory that may work well up to data rates of 700-1000 megabits/second/line is a scheme that converts each single-bit line in the bus into a two-wire, fully differential structure, as diagrammed in two versions in Figure 4. We have been able to demonstrate that many of the signal contaminants that have been observed on differential data lines are so-called common mode phenomena; that is, the same wavefront distortions appear on both "legs" of a differential data line in synchrony with one another, and hence can be removed almost completely by a differential line receiver with good common mode signal rejection characteristics.[18] Extending this finding, we have proposed a design for a differential bus transceiver that does allow multiple signal sources and destinations to be connected to a common "wire" while maintaining high signal transmission rates; this approach preserves the architectural simplicity of bus interconnections inside a high clock rate processor.

If the differential bus approach cannot be made to perform satisfactorily, there is then no choice but to revert to communications structures that use only single sources and single destinations to obviate ringing and reflections on the data lines.[18] However, since the need to route data and instructions between different sections of a processor still remains, some alternate architectural approach must be devised. One such method may be conceptualized as a hub-and-spoke structure,[21] in which every node feeds a set of source lines to and receives a set of destination lines from a central IC or group of circuits; the function implemented within these centrally located chips is a word-wide multiplexer in series with a word-wide output selector, with the two functions separately controllable. Whichever node gains control of this "bus" sets the multiplex function to accept data from the controlling node, and sets the output selector to route data to one

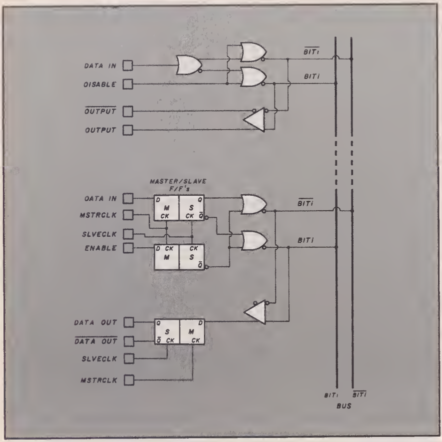


Figure 4. Two versions of a differential transceiver for use in very high data bandwidth bus architectures. Upper transceiver is not pipelined, while lower transceiver is pipelined to buffer out bus line delays in the highest speed systems. The differential structure suppresses signal reflections typical of single-wire busses because these reflections tend to be common mode, and hence are cancelled by the common mode rejection features of the line receivers in each signal transceiver.

or more of the destination nodes. By routing the data in this manner, the single-source, single-or-multiple-destination structure of the hub-and-spoke network minimizes reflections and allows single wire-per-bit data to flow at high speeds, while preserving the topology if not the geography of a bus design. Because long wait times are not required to permit the wavefront reflections to die away, as would be necessary on a conventional bus running at high speed, a hub-and-spoke communications path can operate as fast as or faster than can the bus, even including the additional gate delays through the multiplexer/demultiplexer component(s).

The reader may question why such low level issues have been raised in a discussion of architectures for high-speed GaAs circuits. Note that the use of differential bus structures will have a formidable impact on pin availability on every IC involved with intraprocessor communications, on the internal designs of these circuits, and hence on the partitioning of the entire processor. Similarly, the replacement of a conventional bus with a hub-and-spoke architecture would have a major impact on the mechanisms implemented to resolve "bus" contention, on the firmware developed for the system, and even on the ways in which application programs are executed on the processor. Again, this discussion is but one more example of the way in which the high clock rate nature of GaAs ICs can cause major ripple effects that propagate to the highest architectural levels of a processor design; the would-be GaAs systems designer must become aware of these new problems very early, lest he or she be surprised at a later stage of a new processor design.

Design methodology for GaAs components

Three generic design philosophies are available for the development of high-performance GaAs ICs. The first and best known of these is custom chip design. Throughout the early developmental phases of GaAs digital ICs, custom design has been relied upon heavily to produce the most compact, highest speed, or lowest power components with the best possible fabrication yields; the penalties (as always with custom designs) have been high design costs and lengthy turnaround times. In the rapid turnaround design and fabrication environment typical of many processor design cycles and virtually all military/aerospace design projects, custom design may thus be unacceptable for GaAs ICs except for memory components or microprocessors (such as the DARPA-RISC project described earlier, which *will* result in a custom chip implementation); the design costs in both manpower and computer time are too high, and the design cycles too long, for ICs that may be manufactured in relatively small quantities.

An intermediate approach to rapid turnaround design of ICs is the configurable gate array, in which an array of active "islands" or cells, each composed of a small number of unconnected transistors, resistors, and/or diodes, are created in the lower layers of an IC. The elements of each island can be connected to one another during the final metal layer processes to form any one of a half dozen or more simple gate types, or more complex functions referred to as "macros." This type of active island or cell structure is quite versatile because the desired gate type can always be made available precisely where it is needed on the IC for maximum layout efficiency. Since the active islands are in turn connected to one another to form even more complex functions with only two or three layers of metal lines and one or two via layers on the upper surface of the array,^{2} some types of wafers can be fabricated and stockpiled against the time that they are needed.

Because the total "customization" required for gate arrays requires only the final few layers on the array, chip turnaround from initial design to available parts can be a matter of weeks (provided that the appropriate computer-aided design software packages are available to support the rapid fabrication process). There is at present a strong interest in the provision of a rapid turnaround configurable gate array capability for GaAs digital ICs, which would allow the inclusion of GaAs components in demonstration processors without heavy commitments to custom designs; the lower levels of chip integration and speed performance caused by the rather loose gate packing density of configurable gate arrays will be counterbalanced by the availability of a quick-response chip design capacity in this new material. Considerable development in this regard for GaAs depletion-mode and enhancement-/depletion-mode MESFET gates is presently under way in the research laboratories of several defense contractors.[2,4] Under DARPA sponsorship or with corporate funds, four aerospace corporations, Honeywell, Inc., Rockwell International, Texas Instruments, and McDonnell Douglas, have designed and are now fabricating prototype configurable gate arrays in the size range of approximately 1800-6000 gates,[22-23] with 8000- to 10,000-gate structures appearing feasible by the end of the decade. Finally, at least one civilian-sector company, Tri-Quint, Inc., is developing GaAs configurable gate arrays using GaAs enhancement-/depletion-mode MESFET transistors.[4]

The second possibility for semicustom chip design, and the best way to achieve maximum speed performance short of a custom chip design, is the use of so-called standard height cells, or simply standard cells. In this approach, functional building blocks, or macros, of 5- to 200-gate complexity are custom designed to the transistor level and heavily simulated to verify and guarantee their performance. The macros are carefully selected for their wide applicability to numerous signal processing problems and their ease of configuration in a variety of ways as building blocks to assemble much more complex functions. The individual standard cells are essentially custom designs at all mask levels, but because their layout and function remain constant over time, the individual mask layers for each cell can be stored in a library on computer disk and recalled for repeated use. A number of standard cells are laid out on the chip, with "customized" interconnects designed to coordinate the cells into a complete functional entity. Since logical and analog simulations can be performed with extreme rigor on the custom cells, ultimately this approach provides a very low risk, rapid turnaround method of achieving ICs with high levels of performance. The standard cell approach will eventually be very attractive for the rapid implementation of signal processors exploiting GaAs ICs, in large measure because of the regularity of logic and arithmetic functions employed in signal processors.

The development of design techniques for GaAs-configurable gate arrays is more advanced than for GaAs standard cells, and there are valid technical reasons why this has occurred. The design and verification through computer simulation of the small number of cell types in a gate array is a relatively straightforward and inexpensive task, allowing the fabricators of GaAs circuits to justify this initial cost. If the fabrication process and/or the chip layout change, usually less than a dozen cells, each of one- or two-gate complexity, need be redesigned and resimulated. Conversely, each standard cell is essentially a rather complex custom-designed logical function; every transistor in the function is uniquely specified for its particular task. Thus, every standard cell must be completely simulated, then fabricated and tested, before it can be added to the library of available functions. This process is consumptive of both computer and human effort, and hence is very expensive. If the fabrication technology changes after the functional library has been established, all of the library elements may become nonoperative and may have to be completely redesigned. Because the silicon fabrication technology is currently evolving less rapidly than in the past, the commitment to standard cell libraries is often considered an acceptable risk. In GaAs, however, a stable set of fabrication technologies has only recently begun to evolve; for example, the method of growing GaAs ingots has changed markedly at least three times since 1980 (with an additional modification now in prospect), each time with impacts on the remainder of the IC fabrication processes using these ingots. As a result of these uncertainties, most fabricators of GaAs devices are only now becoming willing to develop and exploit standard cell design and fabrication approaches. However, as the GaAs technology matures, it is expected that greater reliance will be placed upon standard cell libraries, which will yield much more performance than gate arrays but with much less design effort than fully custom components. At least two commercial suppliers of GaAs devices, Gigabit Logic and TriQuint, Inc., have begun to use or investigate the standard cell approach; additional merchant vendors of GaAs digital devices will almost certainly follow suit during the next two years. Finally, AT&T uses small standard cell functions in the layout of GaAs ICs that it fabricates for internal consumption.

Each of the three approaches to the design and fabrication of GaAs ICs has both advantages and deficiencies; as noted earlier, because of the intimate relationships between chip design and system design, the system architect must become familiar with the trade-offs among these various approaches to IC implementation, as well as the trade-offs between the GaAs technologies themselves irrespective of the implementation approach.

Built-in self-test issues for high-speed GaAs ICs

An additional issue, previously the purview of the IC designer, which must now be accounted for by the would-be designer of GaAs processor architectures, is that of operational testing of both the individual ICs and the entire processor once fabricated. Silicon designers, confronted with complexity levels of individual chips exceeding 50,000 gates, as well as the increasing expense of chip testers, are working diligently to identify mechanisms for chips to assess their own performance; by including self-test functions in the very core of the chip design, a higher percentage of the deeply buried gates in the design can be verified. Further, the removal of the financial burden of sophisticated computer controlled testers would in turn decrease the cost of the individual devices. Thus, cost issues are significant drivers in the development of on-chip test logic for silicon devices; however, the testing of these devices could in principle be conducted with externally generated test programs on a modern tester.

Digital GaAs ICs will also require on-chip self-test capabilities; here again, however, the reasons and motivations are somewhat different from those of silicon. Chip complexity is not the issue; GaAs devices will not achieve the 50,000-gate level for some years, and will initially be sufficiently costly that the cost burden of external testing may even be a low percentage of the purchase price of the device. The most critical reason for on-chip test functions in GaAs is that even the lowest clock rate GaAs gate arrays currently in development will operate at 50-MHz clock rates, which are only now being achieved by the most sophisticated commercial IC testers; the next generation of large gate arrays will operate at 200 MHz, while LSI-complexity second-generation standard cell chips can be expected to operate at gigahertz rates. The fastest GaAs devices, though of low complexity, already operate at clock rates above 1 GHz, and may achieve 5- to 8-GHz operation by the end of the decade. Because of its higher purchase cost, GaAs is likely to be used in many applications in which brute-force speed performance is of paramount concern. In such cases, the ability to verify that a chip operates at 10 MHz or even 50 MHz will simply be unacceptable to the ultimate users of the device; if it cannot be guaranteed that each chip will function at its rated speed, the performance of the total system will be in serious doubt. Since commercial IC testers will thus always remain behind the speed capabilities of the devices, the chips must be designed to verify their own performance at full operating speed. Finally, even if the individual chips could be verified at full speed by external testing, once they are incorporated into a system, the same problem arises: system test hardware capable of operating at high speeds is expensive and requires highly trained technical staff in short supply, and the equipment will always be slower than the GaAs-based systems to be tested.

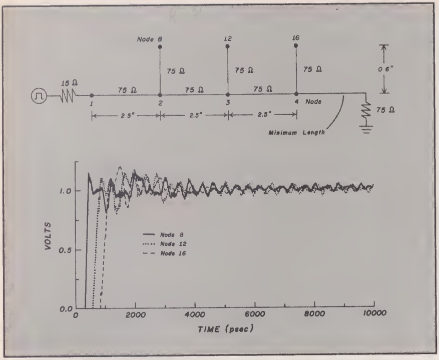


Figure 5. Computer-simulated behavior of a single-source, multiple-destination transmission line logic string connecting digital ICs when exposed to 200-ps risetime pulsed signals. This type of interconnect, the so-called shunt-terminated line, has been used successfully to connect silicon ECL logic circuits (with risetimes of 0.6-2 ns) for more than two decades. However, as risetimes decrease below 200 ps, anomalous ringing occurs because the wider bandwidths of the faster signals reflect repeatedly from the signal string branch points, and from the open circuit stubs at the destination chips (at nodes 8, 12, and 16), creating voltage transients that persist for several nanoseconds. Alternate interconnect approaches common in microwave circuit layouts must be employed to stabilize the interchip signal waveforms.

Fortunately, it is not out of the question to include on-chip test circuits that give a high percentage of gate coverage and that can operate at full chip speeds. Several of the self-test schemes developed for silicon, such as Built-In Logic Block Organization (BILBO), LSSD, Scan-Path, and others (for a comprehensive review of these methods, see reference 24) can make a major contribution to on-chip testing at the full intended clock rate of the component. Although published reviews of on-chip self-test approaches state that optimally at least 20 percent of the chip real estate should be reserved for these functions, small or even "large" GaAs chips may not have a sufficient number of gates to permit such a high percentage commitment of real estate to the self-test function. However, even a commitment of 3 to 5 percent of the available real estate may be sufficient to ensure high confidence in the full speed functionality of the IC; for example, the inclusion of a serial shift-out capability in instruction and data pipeline registers at critical locations in the chip design can be used to examine the states of the chip after each clock cycle or after several clock cycles. Although all gates may not be verified with only 3 to 5 percent on-chip test hardware, coverages above 50 to 75 percent may be achievable in this manner. Finally, it is noteworthy that these same test schemes can, with sufficient prior planning in the design phases of the system, also be configured to allow each chip to retest itself even after incorporation into the processor and to report on its health to a "health manager" function. Here again it should be noted that the design planning for the ICs can no longer be separated from the design planning for the system; the system architect must understand and plan for the problems and constraints of the chip specialist, and conversely.

Packaging and interconnect technology issues for high-frequency GaAs digital ICs

As noted on several earlier occasions, the class of problems best solved with GaAs digital ICs will exploit the ability of these components to operate at system clock rates of 100 MHz and greater with on-chip gate delays of 10-100 ps and off-chip risetimes of 100-250 ps. Since the early days of high-speed silicon ECL components, it has been common practice to employ transmission line interconnects, in combination with resistor termination networks, whenever it was desired to transmit signals between chips whose off-chip risetimes were 2 ns or less in duration; GaAs risetimes are now at least eight times faster than that of early 1970's vintage ECL. Thus, the absolutely mandatory requirement for a controlled impedance circuit board interconnect environment for signals with such extremely short risetimes implies that all aspects of the chip encapsulant and circuit board design be mutually complementary. Both the chip encapsulation and the logic board substrate must be designed to preserve the transmission line environment, from the interfaces between the encapsulant leads and the logic board at least to the perimeter of the die-well inside the chip encapsulant, if not directly to the chip itself. As will be described below, the chip packages and the interconnections on the circuit boards must be treated as high frequency passive analog electrical components connected in series with all of the inputs and outputs of all of the chips in a processor system.

The absence of controlled impedance interconnects in GaAs digital systems will in most cases result in multiple signal reflections that degrade waveform conformations for many nanoseconds, as illustrated in the simulation of Figure 5. This simulation depicts the magnitude and duration of ringing on a *properly terminated* signal string, as might be found on a typical logic board. Presently available chip encapsulants do not provide a suitable electrical environment, and must be redesigned to do so. The stresses that these new high-speed devices will exert on the packaging technology are considerably more severe than those generated by the high-density, low-speed silicon devices.

Example of a high-clock-rate processor: Digital RF Memory

It may be instructive to describe one such very high clock rate system presently under development in our laboratory in order to illustrate the problems in system design, layout, and packaging, which are the principal topics of discussion of this presentation. A Digital RF Memory is essentially an A/D converter generating 2-, 4-, or 6-bit samples of an analog signal at rates of hundreds of megahertz or greater, a serial-to-parallel converter that packs several samples together into a wider word (thereby decreasing the data rate), a

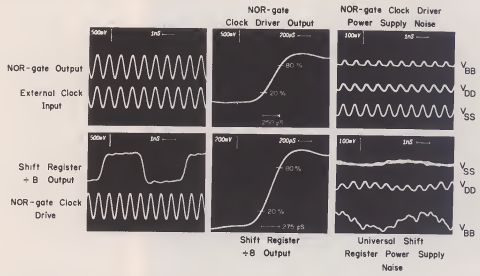


Figure 6. Waveform measurements from selected locations within the high-frequency GaAs front end subsystem of a Digital RF Memory operating at 1.14 GHz. The 250-ps signal risetimes (upper middle panel) did not cause undue amounts of overshoot and ringing on the signal lines (two lower leftmost panels). Sheet resistance and inductance in the power and ground planes, which are the current return paths for the signal strings, were minimized (upper and lower rightmost panels) by providing multiple return paths through the use of 12 power/ground planes in the circuit board, although some residual noise on the power planes may be observed. All such issues must be dealt with at the outset of system design if processors are to operate properly at clock rates above 1 GHz.

read/write scratch pad memory in which the data may be stored temporarily, and a parallel-to-serial converter that speeds up the data rate and feeds the digital inputs of a D/A converter.[9] Since only the very front end and very back end of the system need operate at the highest speeds, it is common practice to treat those sections separately from, for example, the large memory, which can operate at clock rates perhaps 1/8 or 1/16 those of the highest speed sections of the system. In the DRFM fabricated at the Mayo Foundation, the memory section was assembled with commercial ECL components and operated at 125-MHz clock rates. At that frequency, multiple destination gates in the system could be driven from a single source gate; straightforward transmission line termination approaches could be used; the chip carriers and logic boards did not themselves present overly difficult design problems; off-chip signal risetimes were not less than 600 ps; the highest analog frequency components of these signals did not exceed 1-2 GHz; and the signal waveshapes were of good conformation and essentially free of reflections and crosstalk.

It was a much more difficult, though ultimately feasible, task to achieve correct operation in the sections of the system that operated at 1-GHz clock rates; the waveforms of Figure 6 indicate that good signal quality was attained after a number of potentially serious issues were correctly accounted for. Even the slightest mismatches of transmission line characteristics could have caused serious ringing and reflections, which in turn would have destroyed waveform conformations and decreased the signal/noise ratio of the subsystem. All interconnects had to be properly located and treated as transmission lines to prevent the radiation of energy and the pickup of crosstalk signals. The chip carrier packages employed in the project contributed small amounts of unwanted parasitic inductances and capacitances that, if not properly addressed, could have resulted in local oscillations on the signal lines. Studies indicated that, if not accounted for, sheet inductances in the power and ground planes of the circuit board could have seriously impeded the flow of high-frequency AC return path currents, thereby causing instantaneous voltage fluctuations on the power planes at various locations on the circuit board. To minimize this problem, special multilayer circuit boards were designed that contained 12 power/ground planes to provide multiple shunt paths for the flow of high-frequency return-path currents; in addition, every chip location was liberally supplied with RF decoupling capacitors. Finally, the highest speed signal lines were not permitted to have more than two destinations, thereby decreasing the severity of wavefront reflections and increasing the effective speed of the interconnects. To achieve this goal, logical chip output bits were replicated in several instances. A very considerable effort was expended in the creation of a processor architecture that supported a flowthrough path for the highest speed data as it emerged from the A/D converter and as it entered the D/A converter. In this system, a flowthrough design could be achieved; in other architectures, the designer may not be as fortunate.

This experiment underscored the importance of:

- flowthrough designs and extremely short interconnect lengths for at least the highest speed data and clock paths;
- the careful selection of logical and physical input and output signals on each chip;
- a coordinated system layout concept, in which every IC part type was carefully designed to function at full speed with all the others;
- the execution of computer simulations and timing verifications of all of the individual part types separately and operating together;
- the use of high-frequency models for the GaAs gates and transistors that accurately represent the AC behavior of the actual structures; and finally,
- for the need, in most cases, to develop much higher quality packaging at the chip carrier and circuit board levels than has been available heretofore.

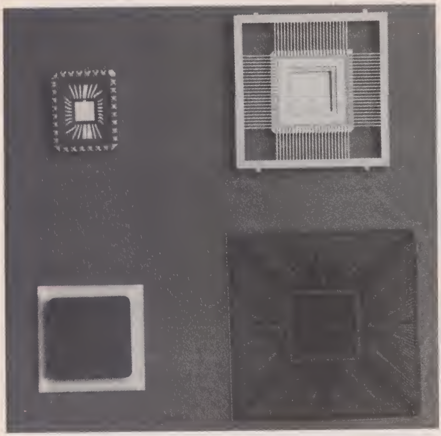


Figure 7. Four generations of controlled-impedance chip carriers designed at Mayo Foundation for GaAs chips. All have 50-ohm or 75-ohm signal leads, dedicated leads and/or internal ground/power planes for ripple-free power delivery, and locations for microwave decoupling capacitors. Clockwise from upper left: a 28-lead, 400 × 500-mil cofired alumina leadless carrier with contacts on 50-mil centers (1981); an 88-contact (64 signal, 24 power/ground) 500 × 500-mil cofired alumina leaded/leadless chip carrier with 20-mil center contacts (1984); a 216-lead (192 signal, 24 power/ground) 1200 × 1200-mil alumina leadless carrier with 20-mil center contacts, shown in film format (1986); a 216-contact (192 signal, 24 power/ground) 650 × 650-mil thin-film-on-BeO leadless minicarrier with 10-mil center contacts (1986). The minicarrier includes 192 thin-film resistor terminators. Note the 8:1 increases in interconnect density between oldest and newest designs; carrier size has remained roughly constant.

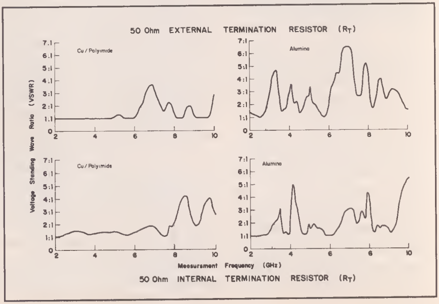


Figure 8. Frequency domain performance of two chip carrier packages designed to encapsulate high-clock-rate GaAs digital ICs. The plots display voltage standing wave ratio (VSWR) as a function of frequency. VSWR values of less than 2 represent the range of satisfactory performance; the polyimide package is usable to signal bandwidths of 6-8 GHz, while the alumina package is usable to 3 GHz. The packages must be conceptualized as analog microwave elements connected in series with the digital ICs that they contain. These analog microwave effects must be accounted for in the planning of the layout and partitioning of high-clock-rate signal processors fabricated with GaAs devices.

It must be reiterated that the design and fabrication of a system that will operate correctly at 1- to 2-GHz system clock rates is extremely difficult. The issues discussed herein are sufficiently complex and intertwined that appropriate solutions for them must be developed by both the system architect and the chip designer working together; otherwise, the final product can be nearly guaranteed not to work as intended, or perhaps at all.

The foregoing discussion of the problems encountered in one example high-clock-rate processor project is intended to forewarn the systems designer of the numerous issues with which he or she will be confronted. The remainder of this presentation will address several of the most critical packaging and component interconnect issues in greater detail, under the assumption that the GaAs ICs will experience clock rates above 100 MHz, will have moderate pin counts (24-200), and will generate short risetimes (less than 500 ps).

Intercomponent connection problems

Chip-to-chip interconnect delays must be minimized even when on-chip designs employ multiple gate stages or implement iterative algorithms. Although it has been noted in several published papers that a deliberate degradation of the off-chip risetimes would minimize the interconnect problem, overall system performance would also be reduced. Computer simulations of the combined wire delay and risetime effects for a given interconnect length and logic board dielectric constant have demonstrated that risetimes as low as 100-200 ps result in a substantial reduction in effective interconnect length. For example, for a logic board dielectric constant of 2.3, a change in the designs of off-chip drive transistors that has the effect of altering the signal risetime from 1 ns to 200 ps is equivalent to a saving of 6.2 inches of interconnect in every logic net on the circuit board.[18] However, even assuming terminated transmission line interconnects, each off-chip drive transistor and its associated line terminator will dissipate 20-80 mW, which in a high I/O count IC can result in unacceptably large on-chip power dissipation. To minimize this problem, two separate but complementary concepts might be explored. First, significant reductions in power dissipation in the chip periphery may be achieved by employing either bit-serial or nibble-serial intercomponent communications. The data operands may be reassembled on-chip to full width over several system clock cycles, or the operands themselves may be processed bit or nibble-serially using iterative algorithms or bit-decomposed algorithms, as described earlier and discussed in references 19 and 20.

Alternately, a re-examination of off-chip electrical transmission protocols might yield an approach with sufficiently high noise margin at lower logic voltage swings, thus decreasing the power dissipated both in the driver transistor and in its line terminator. While silicon TTL components use a 3- to 5-V logic swing, silicon ECL has successfully employed 800-mV logic amplitudes. Provided that the AC noise margin is sufficiently high for the logic family, small-amplitude logic swings can be used without concern for system robustness. Logic swings as low as 500-600 mV appear feasible in a few of the GaAs technologies with inherently high AC noise margins, and are actively being explored through design and computer simulation studies by Mayo Foundation and McDonnell Douglas Microelectronics Center for the GaAs EJFET technology.

If a critical data signal or the clock signal on a circuit board must be kept free of contamination, it is sometimes necessary to transmit the signal differentially from its principal source to its principal destination, receive the signal with a differential line receiver, and then retransmit the signal for short distances as a single ended signal. In the case of critical clock waveforms, the additional precaution is taken of feeding the signal into the center of the board, then using several tiers of differential distribution, with a final conversion to a single ended waveform that is fed to only a few components each at numerous locations around the logic board. As noted earlier, the number of destination gates driven by each source gate must be constrained to four or even less; furthermore, because of high-frequency ringing phenomena, conventional single wire bus-type interconnect structures may not work at all. Either the busses will have to be removed from the architectural design entirely, or a fully differential data path may have to be employed along with special differential line transceiver components to drive onto and receive from the bus or, as a final resort, a hub-and-spoke data distribution network may be required, as described earlier.

Design issues for chip carrier packages and logic boards for GaAs components operating above 100 MHz

The electrical characteristics of the chip carriers used to package GaAs ICs can substantially limit the effective performance available from the highest speed versions of these chips. In general, the DC resistance of the signal leads in the encapsulation is of little consequence, since DC resistance values less than half an ohm are the rule. At high frequencies, however, signal lead parasitic inductive and capacitive reactances are the primary causes of degradation in high-speed chip performance. Although the circuit board interconnects must be designed as transmission lines whenever signal risetimes are less than 1-2 ns, the shunt capacitances measured at the inputs of the encapsulant signal leads always degrade the performance of the board interconnects by creating a local decrease in transmission line impedance. Chip packages connected to a transmission line usually create a local decrease in line impedance, which causes wavefront reflections to propagate to and fro along the signal string. Such reflections are always in the direction of the logic threshold (defined as the "negative" direction), and hence decrease signal noise margins to some extent. If the shunt capacitances are large enough, their effects can be detected even in silicon ECL systems operating at 100-150 MHz.

At frequencies above 100 MHz, the signal paths in the chip carriers, the chip wire bonds, and the circuit board traces must be included in the simulations of subsystem performance; further, these simulations must be conducted both in the digital timing domain and also in the analog electrical sense to completely model the behavior of these elements. Topics previously considered to be the responsibility of the radio frequency (RF) analog microwave specialist in the design of monolithic microwave integrated circuits (MMICs) are now rapidly becoming the concern of the digital systems designer as well. New versions of chip carrier packages intended for GaAs circuits are designed to provide a shielded cavity, transmission line interconnect environment for the ICs, and are simulated using microwave simulation programs. The chip packages should be fabricated with two or more layers of internal ground and power supply planes much like those found in a multilayer printed circuit board, with integral thin film termination resistors, and with integral RF decoupling capacitors to control voltage fluctuations on the internal power supply planes of the carrier. For each successive generation of packages, the carriers must become smaller, with tighter interlead spacing, finer metal line geometries, and improved crosstalk control. Figure 7 depicts several successive generations of these carriers designed at Mayo Foundation for digital GaAs ICs; see the legend for further details. The as-fabricated packages are tested on analog vector network analyzers; the most critical package operating parameters are considered to be voltage standing wave ratio (VSWR), and RF crosstalk (see, for example, Figure 8).

Great care should be also used in the design of logic boards for high-clock-rate GaAs circuits. Test coupons containing examples of various types of interconnects are designed, laid out, and fabricated in the same materials from which the final circuit boards will be manufactured. The coupons are then also tested using analog microwave equipment, and the test results are compared with those derived from the computer models. Disparities between the measured and simulated data are resolved; thereafter, the simulation results are used as design guides during logic board layout. In the near future, the operating parameters of every interconnect in an as-laid-out logic board will be verified by computer simulation to ensure satisfactory high-frequency performance, including VSWR, forward and backward crosstalk, and wavefront reflections (see Figures 5 and 8).

As the clock rates of GaAs-based processors increase beyond 2 GHz, a major change must occur in the traditional method of packaging digital ICs. No matter how carefully designed, single chip or even multichip carriers will extract too great a performance penalty from the ICs; impedance mismatches at the chip carrier/logic board interface will always be difficult to prevent completely, and unavoidable time delays will occur as the signal propagates into and out of the carrier. Ultimately, methods must be identified to bond the chips directly to the logic board substrate; the boards themselves will have to support at least two levels of interconnects of transmission line quality, with lines on 4-8 mil centers to match the contact pad spacing on the chips them-

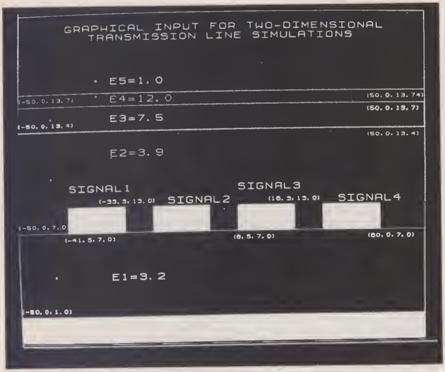


Figure 9. Computer terminal display from a CAD software tool that allows a design engineer to simulate electromagnetic and transmission line behavior of complex metal-dielectric sandwiches; these sandwiches, shown in cross section, are typical of multilayer circuit boards and advanced chip carriers. The designer specifies locations of signal and ground conductors, ground and power planes, and the dimensions of the structures and the dielectric constants of the insulating layers, employing color coding throughout the diagram. The diagram is then parsed in a postprocessing step to extract necessary parameters for electromagnetic simulation. Results are overlaid on the graphical input, including line impedance, mutual and self-inductances and capacitances, and forward and backward crosstalk values.

selves. Wire, ribbon, or tape automated bonds (TABs) will form the electrical connections between the chips and the logic boards. The technology for these substrates will undoubtedly be exotic; however, less sophisticated approaches are unlikely to allow signal processors to operate at the clock rates of which the GaAs components themselves will be capable.

Finally, it must be noted that the accuracy and flexibility of the computer codes employed for the modeling of the high-frequency electrical environment of chips, chip packages, and logic boards must be significantly improved. Computer simulations based on the century-old telegrapher's equations and on relaxation methods must be replaced by the much more powerful theoretical methods that have been developed for both two- and three-dimensional metal/dielectric structures during the past decade (see, e.g., reference 25); these methods are more flexible and accurate, and execute faster on a computer than the older approaches. Based on the method discussed in reference 25, we have developed both two- and three-dimensional computer codes that can simulate the electromagnetic behavior of multilayer IC packages and logic boards, thereby providing chip and package designers with valuable data that aids in the design of these structures. By providing graphically oriented interfaces between the simulation codes and the designers, a user-friendly environment is created that allows the designer to experiment rapidly with various combinations and dimensions of conductors and dielectric sandwich structures until an optimum design is achieved (Figure 9).

The design of high-performance signal processors with GaAs ICs must be a multidisciplinary task, at a level of integration uncommon and probably unnecessary for silicon-based systems. No longer will the system designer be able to concentrate on the architecture of a processor with the certain knowledge that chip design and simulation, packaging and interconnection layouts, circuit board design, and a host of other central issues will be "cleaned up" by specialists in these narrow fields. The design of a GaAs processor that works properly at full speed must be a well-integrated team effort involving representatives from all of these disciplines. The failure to establish and nurture a strong team-oriented effort, with all contributors sharing a working knowledge of the responsibilities of the other team members, is virtually a guarantee of poor performance of the ultimate product, and even of complete collapse of the entire project. This article has attempted to illustrate several of the more crucial areas of concern; however, there is much more to be learned by all workers in this field before the new set of GaAs technologies is completely reduced to practice. □

Acknowledgments

This research was supported in part by contracts MDA903-84-C-0324 and F29601-84-C-0016 from the Defense Advanced Research Projects Agency. The author wishes to thank L. Krueger, S. Karwoski, S. Zahn, D. Endry, C. Treder, G-W. Pan, M. Samson, C. Bates, J. Ryan, B. Shamblin, and T. Volkman, all of the Mayo Foundation, for invaluable technical assistance; S. Roosild and S. Karp, of Defense Advanced Research Projects Agency; J. Kesperis, of the US Army LABCOM, for advice, encouragement, and helpful discussions; T. Webb and J. Spangler, of McDonnell Douglas Microelectronics Center, for fabrication of the 216 pad experimental chip carrier mockup shown in Figure 7; and E. Doherty and S. Richardson, of the Mayo Foundation, for preparation of text and figures.

References

1. S. Long et al., "High-Speed GaAs Integrated Circuits," *Proc. IEEE,* Vol. 70, No. 1, Jan. 1982, pp. 35-45.
2. R. N. Deming et al., "A Gallium Arsenide Configurable Cell Array Using Buffered FET Logic," *IEEE J. Solid State Circuits,* Vol. SC-19, No. 5, Oct. 1984, pp. 728-738.
3. W. Yuan et al., "GaAs Heterojunction Bipolar 1K Gate Array," *Proc. IEEE 1984 Int'l Solid-State Circuits Conf.,* pp. 42-43.
4. A. Rode, T. Flegel, and G. LaRue, "A High Yield GaAs Gate Array Technology and Applications," *1983 IEEE Gallium Arsenide IC Symp.,* pp. 178-181. IEEE Publ. #83CH1876-2.
5. S. S. Pei, N. Shah, and C. Tu, "Bell Labs Transistor Sets a Speed Record," *Science,* Vol. 231, No. 4736, Jan. 24, 1986, p. 340.
6. T. Ohshima et al., "A Self-Aligned GaAs/AlGaAs Heterojunction Bipolar Transistor With V-Groove Isolated Planar Structure," *Proc. IEEE 1985 Gallium Arsenide Integrated Circuit Symp.,* pp. 53-56. IEEE Publ. #85CH2182-4.
7. H. Yuan, "GaAs Bipolar Gate Array Technology," *Proc. IEEE 1982 Gallium*

Arsenide IC Symp., pp. 100-103. IEEE Publ. #82CH1764-0.

8. R. O. Grandin, W. Porod, and D. K. Ferry, "Delay Time and Signal Propagation in Large-Scale Integrated Circuits," *IEEE J. Solid State Circuits,* Vol. SC-19, No. 2, Apr. 1984, pp. 262-263.
9. B. K. Gilbert et al., "Design and Fabrication of a Digital RF Memory Using Custom Designed Integrated Circuits," *Proc. IEEE GaAs IC Symp.,* 1985, pp. 173-176. IEEE Publ. #85CH2182-4.
10. H. T. Kung, "Why Systolic Architectures?," *Computer,* Vol. 15, No. 1, Jan. 1982, pp. 37-46.
11. T. J. Fountain, "The Development of the CLIP7 Image Processing System," *Pattern Recognition Letters,* Vol. 1, pp. 331-339.
12. K. Preston and M. J. B. Duff, *Modern Cellular Automata, Theory and Applications,* Plenum Press, New York, 1984. ISBN 0-306-41737-5.
13. E. E. Swartzlander, Jr., B. K. Gilbert, and I. S. Reed, "Inner Product Computers," *IEEE Trans. Computers,* Vol. C-27, No. 1, Jan. 1978, pp. 21-31.
14. B. K. Gilbert et al., "Rapid Execution of Fan Beam Image Reconstruction Algorithms Using Efficient Computational Techniques and Special-Purpose Processors," *IEEE Trans. Biomedical Engineering,* Vol. BME-28, No. 2, Feb. 1981, pp. 98-116.
15. A. Kazuyoshi, K. Kurumada, M. Hirayoma, and M. Ohmori, "GaAs 1 K-Bit Static RAM With Self-Aligned FET Technology," *IEEE J. Solid-State Circuits,* Vol. SC-19 , No. 2, Apr. 1984, pp. 260-262.
16. R. Eden, "Capacitor-Diode FET Logic (CDFL) Circuit Approach for GaAs D-MESFET ICs," *Proc. 1984 IEEE GaAs IC Symp.,* pp. 11-14. IEEE #84CH2065-1.
17. T. Gross, *Core Set of Assembly Language Instructions for MIPS-based Microprocessors,* Version 3.0, April 2, 1986.
18. B. K. Gilbert, "Packaging and Interconnection of GaAs Digital Integrated Circuits" in N. G. Einspruch and W. R. Wisseman (eds.), *VLSI Electronics Microstructure Science,* 1985, Academic Press, New York, Vol. 11, pp. 289-331.
19. J. V. McCanny and J. G. McWhirter, "Implementation of Signal Processing Functions Using 1-Bit Systolic Arrays," *Electronics Letters,* Vol. 18, No. 6, March 1982, pp. 241-243.
20. J. G. McWhirter et al., "Multibit Convolution Using a Bit Level Systolic Array," *IEEE Trans. Circuits and Systems,* Vol. CAS-32, No. 1, Jan. 1985, pp. 95-99.
21. B. K. Gilbert et al.,"A Real-Time Hardware System For Digital Processing of Wideband Video Images," *IEEE Trans. Computers,* Vol. C-25, No. 11, Nov. 1976, pp. 1089-1100.
22. T. T. Vu et al., "Low-Power Logic Circuits and 2K Gate Array Using GaAs Self-Aligned MESFETs," *IEEE J. Solid State Circuits.* To be published.
23. G. L. Troeger and J. K. Notthoff, "A Radiation-Hard Low-Power GaAs Static RAM Using E-JFET DCFL," *Proc. IEEE 1983 Gallium Arsenide IC Symp.,* pp. 78-81. IEEE Publ. #83CH1876-2.
24. T. W. Williams, "Design for Testability—A Survey," *Proc. IEEE,* Vol. 71, No. 1, Jan. 1983, pp. 98-112.
25. C. Wei et al., "Multiconductor Transmission Lines in Multilayered Dielectric Media," *IEEE Trans. Microwave Theory and Techniques,* Vol. MTT-32, No. 4, Apr., 1984, pp. 439-449.

Barry K. Gilbert is a staff scientist in the Dept. of Physiology and Biophysics of the Mayo Foundation in Rochester, Minn. He is also the director of the Special- Purpose Processor Development Group at the Mayo Foundation.

His research interests include the development of algorithms for the real-time analysis of wide bandwidth image and signal data; the design of specialized signal-processing computers to execute these tasks; the development of CAD tools to allow the timely design of high-complexity digital signal processors; and the advancement of high- performance IC technologies, such as GaAs, that can be used to assemble very-high-performance signal processors.

Gilbert received his PhD degree in physiology and biophysics from the University of Minnesota.

Barbara A. Naused joined the Mayo Foundation's Special-Purpose Processor Development Group in 1981; she has experience with signal processor architecture systems and GaAs IC circuit design. She has conducted comparative reviews of microprocessor architectures, and worked on their hardware and software designs.

She is completing a master's degree in electronics engineering from the University of Minnesota.

Daniel J. Schwab has been a member of the Mayo Foundation's Special-Purpose Processor Development Group since 1979. While working for the group, he has specialized in the layout, fabrication, and testing of high-speed digital processors that incorporate high-speed subnanosecond emitter-coupled logic, gallium arsenide digital ICs, and conventional TTL logic.

Schwab attended the Mankato Vocational-Technical School from 1974 to 1976, where he specialized in instrumentation and control technology. He is presently completing his BS degree in electrical engineering at the University of Minnesota, Minneapolis.

Rick L. Thompson joined the Special-Purpose Processor Development Group in late 1984, after graduating from Purdue University with a BS degree in electrical engineering technology, with subspecialties in analog and digital systems. He has designed a number of GaAs IC circuits of up to 6000-gate complexity, and has worked extensively with CAD software for GaAs processors.

Prior to beginning work on his BS degree, Thompson spent four years in the US Air Force in Germany as a telecommunications control specialist.

Readers may write to Gilbert at the Medical Science Bldg., Mayo Clinic, Rochester, MN 55905.

CALL FOR PAPERS

24TH ACM/IEEE DESIGN AUTOMATION CONFERENCE®

DAC 87 is the premier conference devoted solely to the field of Design Automation. All aspects of the use of computers as aids to the design process from conceptual design through manufacturing are included. The conference offers both a high quality technical program of fully refereed papers and the opportunity to meet vendors of new CAD hardware and software in an exhibit area. Four session types are included: regular paper sessions, short paper sessions, panels, and tutorials.

REQUIREMENTS FOR SUBMISSION OF PAPERS

Authors should submit their papers to the program chairman no later than **November 14, 1986** . Each submission should include one cover page and five copies of the complete manuscript. The one cover page should include:

- Name(s), affiliation(s), complete address(es), identification of principal author and telephone number.
- The following signed statement: "All appropriate organizational approvals for the publication of this paper have been obtained. If accepted, the author(s) will prepare the final manuscript in time for inclusion in the conference proceedings and will present the paper at the conference."

The five copies of the complete manuscript should each include:

- Title and abstract page: title of paper, 60 word abstract indicating significance of contribution, and an ordered list of number(s) from the categories at the right that most closely match the content of the paper. To permit a blind review, do not include name(s) of author(s) on this page. If accepted, the 60 word abstract will be printed in the advance program.
- The complete text of the paper in English, including illustrations and references, not exceeding 5000 words. The length of the paper is considered in the review process.

Notices of acceptance will be sent to the principal authors by February 13, 1987. Authors of accepted papers must sign a copyright release form.

TOPICS OF INTEREST

Authors are invited to submit original technical papers describing recent and novel research, development, or product developments in all areas of design automation. Topics include, but are not limited to:

1. Simulation
2. Testing, Diagnosis
3. IC Layout
4. Silicon Compilation
5. PCB Placement and Routing
6. Layout Verification, Design Rule Checking
7. Timing Verification
8. PLA and Logic Level Design Aids
9. System Level Design Aids
10. Design Systems and Methodologies
11. Hardware Description Languages
12. Special Purpose Hardware for DA
13. Human Factors in DA
14. Databases
15. IC Fabrication
16. Analog Circuits
17. Software Engineering in DA
18. Expert Systems

PANELS, TUTORIALS, AND SPECIAL TOPIC SESSIONS

Proposals for topics of Panels and Tutorials should be submitted to the program chairman no later than December 12, 1986. Each proposal should not exceed two pages and include the description of the topic, structure of the session, participants, etc. Contact should have been made with the participants in advance.

Proposals for Special Topic Sessions should be submitted along with the papers to be included in the session and should describe the theme of the session. Submit the papers as described on the left. These proposals should also be submitted by November 14, 1986.

PROGRAM CHAIR

MP Associates, Inc.
ATT: Donald E. Thomas
Program Chair 24th DAC
7366 Old Mill Trail, Suite 101
Boulder, Colorado, 80301
For information call: 303-530-4333

acm SIG DA

THE COMPUTER SOCIETY OF THE IEEE
DATC

MIAMI BEACH CONVENTION CENTER JUNE 28 – JULY 1, 1987

A GaAs E/D MESFET demonstration wafer that has digital logic (1-micron two-level metal). (Photo courtesy of RCA Advanced Technology Laboratories.)

Issues of Importance in Designing GaAs Microcomputer Systems

Veljko Milutinović, Alex Silbey, David Fura,
Kevin Keirn, and Mark Bettinger
Purdue University

Walt Helbig, William Heagerty, Richard Zieger,
Bob Schellack, and Walt Curtice
RCA Corporation

Many more factors are proving to be drivers of GaAs VLSI architecture than was ever the case with silicon. However, the possibility that GaAs may make possible throughput five times greater than that achievable with silicon continues to encourage research.

In 1984, RCA began projects that would lead to the development of equipment constructed with GaAs digital VLSI technology. Continuing the approach taken in developing VLSI design capability for CMOS and CMOS/SOS, three technique-development projects were undertaken and, in this case, a fourth was added. The three previously followed were standard logic-cell circuit design, CAD program development for automatic chip placement and routing, and one or more chip architecture designs. The new technique-development project concerned chip packaging and interconnect for GaAs VLSI technology that would operate in the 100-MHz to 400-MHz range.

Initially, the four projects were all company-sponsored efforts, and the architecture example chosen was an 8-bit reduced instruction set computer (RISC) designed along the lines of a Stanford University multiprocessor without interlocked stages (MIPS). Subsequently, a contract was obtained from the US Government to direct similar architecture design efforts toward the design of a 32-bit computer (designed as a RISC) that would have a 200-MIPS throughput.

In pursuing both the 8-bit and 32-bit computer efforts, it was immediately recognized that there were a number of limitations associated with GaAs digital circuits that would affect the design and assembly of such computer systems. These limitations made it necessary (1) to reevaluate many technique proposals that were made during the early days of silicon VLSI development and (2) to develop new techniques to solve old problems.

For example, some of the ideas that were considered for GaAs architectures were silicon technology designs that had been discarded because they did not provide enough benefit. However, because of the differences in GaAs and silicon technology parameters, some ideas that are unusable or rarely used in silicon designs seemed to make sense for GaAs. One example is migration of hardware functions back into software.[1-2]

The difference between silicon technology parameters and GaAs technology parameters is much greater than the dif-

0018-9162/86/1000-0045$01.00 © 1986 IEEE

ference between the parameters of two different silicon technologies. Most designers already know that when minor technology changes occur, the basic design strategy remains the same (for example, when the instruction pipeline depth is forced to change from three to four because of a slight increase in the CPU speed). But when one or more parameters change from a half to a full order of magnitude (as is sometimes the case when changing from silicon to GaAs parameters), the strategy must change as well. For example, the cache memory was created to fill a sudden gap between CPU speed and memory speed caused by technology changes in the 1960's.

Technology changes affect both processor and system design. The processor design issues were covered by Milutinović, Fura, and Helbig[3]; here we cover system design issues. Although these two types of issues are essentially different, they are highly interrelated.

Silicon CMOS and E/D MESFET GaAs technologies. Different GaAs design processes are characterized by different computer design parameters, and the same is true of the various silicon technologies. Therefore, it is important to keep in mind that this article deals only with (1) the E/D MESFET process currently being used by RCA[4] and (2) with the CMOS silicon process developed at RCA. ("E/D MESFET" stands for enhancement/depletion-mode metal semiconductor field effect transistor.)

One of the most critical differences between CMOS silicon technology and E/D MESFET GaAs technology is that the GaAs wafers have a higher density of defects. This results in a very low chip yield and indirectly limits individual chip area. Whereas some CMOS silicon designs *reliably* have up to approximately two hundred thousand transistors on a single chip, the highest *reliable* level of integration achieved so far in GaAs technology has been below thirty thousand transistors.[3]

One of the other crucial differences between CMOS silicon technology and E/D MESFET GaAs technology is the ratio of off-chip memory access speed to on-chip memory access speed. Because the penalty for accessing off-chip memory is much higher in GaAs technology than in silicon technology,[3] a restructuring of the silicon CMOS memory hierarchy to minimize the number of off-chip accesses, or, alternatively, to minimize the penalties for going off-chip, is necessary.

Another major difference between CMOS silicon and E/D MESFET GaAs is the more limited fan-in/fan-out capability of GaAs. Whereas an experienced silicon technology designer can choose to specify a 3-input NAND gate in his designs, the GaAs technology designer simply may not have such an option. Furthermore, each extra output on a GaAs NOR gate results in a relatively large increase in switching time.

To make evaluations of the many proposals, extensive simulations were felt to be necessary. For this purpose, RCA chose to employ the assistance of Purdue University. Both the RCA group and the Purdue group wrote hardware-description models and ran simulations of various design approaches and construction parameters. This article is a summary of what was learned in the specific areas of interconnection technology, as well as in a GaAs computer system's memory, arithmetic coprocessing, and I/O.

Memory design in the GaAs environment

We found that memory-related design issues are affected by GaAs more than other system-design issues (more than, for example, I/O, arithmetic, or coprocessing).

Cache-based memory. The design of cache memories is strongly influenced by the characteristics of the underlying technology.

Technology characteristics relevant for cache design. One extremely important difference between silicon technology and GaAs technology is the much lower transistor-count capability of GaAs memory chips. Because of this, GaAs caches will generally be much smaller than their silicon counterparts.

The dissimilar ratios of off-chip memory access speed to on-chip memory access speed is another important difference between the technologies. The higher penalty for off-chip access in a GaAs environment requires a careful structuring of the memory hierarchy to minimize the penalty.

The lower chip yield of current GaAs technology also necessitates a different approach for GaAs cache design. Techniques that increase chip yield are more appropriate for designing GaAs memory chips than for designing silicon memory chips, which have a relatively high yield.

Finally, the limited fan-in/fan-out capability of GaAs gates can impact cache design, too. Implementing the capability for wide associative searches (for associative cache memories) is expensive in GaAs. Additional chip area is required to implement the fan-in/fan-out trees required by associative logic.

Cache hierarchy in the GaAs environment. Because of the speed at which information is transferred between a GaAs processor and main memory, examination of a second-level cache deserves consideration.[5]

The composition and physical placement of the second-level cache are extremely important to ensuring the timely transfer of information between the first-level cache and the GaAs processor. To achieve maximum memory density at a suitable speed, the first-level cache should consist of GaAs chips. Also, to minimize the damaging effect of propagation delays, the first-level cache should be placed on the processor package.[3]

More flexibility is possible in both composition and physical placement when a second-level cache is added to a GaAs coprocessor and a first-level cache. GaAs technology may not be appropriate for the second-level cache, unless it is absolutely necessary for reasons like radiation hardness. The relatively large size of this cache will require that the cache use a large number of the relatively sparse GaAs chips, and this use will create a large distance between the GaAs processor and the farthest memory chips. Higher density silicon chips may actually supply faster access times because of their closer proximity to the GaAs processor, even though their raw speed is lower than that of the GaAs memory chips. For this reason, careful study is required to select a configuration involving a second-level cache.

Fetch policies in the GaAs environment. Static prefetching is of considerable interest in helping reduce the miss ratio in a GaAs microprocessor cache. *Static prefetching* is done at compile-time, and it involves predicting the blocks most likely to be needed to be moved into the cache. The compiler is able to do this more intelligently than the hardware because of its knowledge of the semantics of the code. For example, the compiler can recognize an unconditional branch in a high-level language code and prefetch the target of that branch. The small cache size, common in GaAs systems, means that static prefetching should be used sparingly so as to avoid polluting the cache with data that is never referenced. If static prefetching is used only when the prefetched block is certainly (or nearly certainly) going to be referenced, the danger of memory pollution is lessened.

In conclusion, for a GaAs microprocessor system, the static prefetch method has three advantages over the *dynamic prefetch* method, which is carried out at run time.

- First, static prefetching can be "smart." Memory pollution can be avoided or reduced by using knowledge of program semantics to make accurate prefetch decisions.

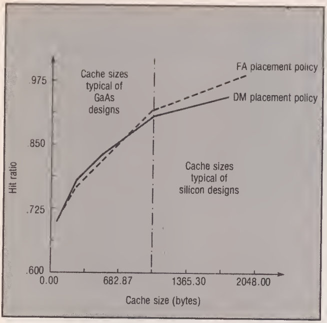


Figure 1. A comparison of the hit ratios achieved by the direct mapped (DM) placement policy and the full associative (FA) placement policy.

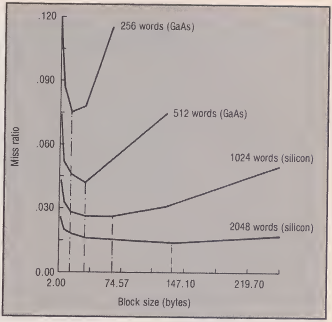


Figure 2. The relationship between miss ratio and block size as plotted for four different cache sizes.

- Second, static prefetching does not increase the hardware complexity of the cache.
- Third, the miss ratio decrease that occurs when static prefetching is used results without a corresponding increase in the traffic ratio (that is, in the total references versus program-generated references).

However, the burden in static prefetching is moved to compile-time; the effect this has on compiler complexity will be described in Milutinović et al.[1]

Placement policies for the GaAs environment. Placement policies regulate the location of blocks after they are transferred from main memory into the cache as a result of the fetch policy. There are three common placement policies: *direct-mapped* (*DM*), *set-associative* (*SA*), and *fully associative* (*FA*) (in order of increasing hardware complexity). The criteria for selecting a placement policy include the miss ratio, the complexity of the hardware, and the access time of the cache.

- *The effect of placement policy on net cache area.* The FA and SA placement policies allow more flexibility than the DM policy in the placement of individual data blocks. However, along with the increased flexibility of the FA and SA policies goes the need for a *replacement policy* to decide which block to replace in the cache when the cache is full. The FA and SA placement policies require that some overhead bits be used to store usage information on each block. When the commonly used least recently used (LRU) replacement policy is invoked, the amount of overhead is $\log_2 N$ bits per block, where N is the number of blocks in an FA cache and N is the "set size." These overhead bits can be used for net data storage when a DM placement policy is used, since the DM policy requires no replacement policy. Another advantage of the DM placement policy is that the same cell can be used for memory and tags. Since no special associative search between tags and requests by processor memory for blocks has to be made, there is no reason to have special cells for the tag bits. The net storage area gained by using the DM placement policy could be significant, since the overall number of bits in a GaAs cache is restricted by transistor count limitations.

- *The effect of placement policy on hit ratio.* In general, the overall ranking of these three placement policies in order of increasing hit ratios is DM, SA, then FA. However, for small caches, it is not always true that the FA placement policy results in a higher hit ratio than the DM policy. Results from experiments with general-purpose code traces on a 32-bit minicomputer indicate that the DM policy performs comparably to the FA policy for cache sizes up to 1024 bytes.[6] The results are shown in the Figure 1 graph, in which the solid line indicates the DM policy and the broken line indicates the FA policy. For small caches, the DM placement policy provides equal or superior hit ratios.[6] The hit ratios are equal at cache sizes of about 900 bytes, and remain close to each other until cache size reaches about 1K byte. Between 1K byte and 8K bytes, the FA placement policy provides a considerably better hit ratio. At sizes larger than 8K bytes, the placement policy has little effect on the hit ratio, since the ratio asymptotically approaches 1.0 as the cache size increases. Figure 1 indicates the cache sizes for which GaAs is practical (those to the left of the dotted-dashed line).

Block size in the GaAs environment. After net cache size, block size probably has the largest effect on the performance of a cache memory. The *block size* is the number of bytes of cache memory associated with each address tag in the cache.

Different block sizes can have dramatically different influences on the miss ratio. As already indicated, the miss ratio is even more important in GaAs than in silicon because of the huge penalty in computing speed caused by a miss in a GaAs cache.[3] Results from experiments show that the optimum block size for smaller caches is smaller than the optimum block size for larger caches.[7] However, even in a large cache, the block size must be restricted because of the danger of memory pollution. Figure 2 shows the experimentally derived relationship between miss ratio and block size for four cache sizes.[7] Note that the minimum block size should be larger than or equal to 1 + the pipeline branch latency.

Storing data. There are two common techniques for storing data: write-through and copy-back. These either (1) immediately transfer modified data words to the main memory (*write-through*) or (2) transmit the entire modified data block to main memory when a miss occurs and the block is selected to be replaced (*copy-back*).

In the write-through implementation, all write instructions result in the immediate transmittal of the datum to main memory. Thus, the processor must wait (if there are no buffering facilities) for the write to complete before it can continue processing. In a GaAs microprocessor system, however, buffering may not be needed, since, as we discuss later, pipelined memory is an efficient solution to the problem of long off-chip latency. With a pipelined memory, there is no way that a read request and write request can get out of sequence. The pipelined memory also eliminates the wait for the completion of the write. Write-through also ensures that an updated copy of the cache contents always exists in main memory.

The copy-back method uses a "dirty bit" to indicate blocks in the cache that have been modified. When the replacement policy decides that a modified block is to be replaced, then that block has to be written back to main memory before a new block can take its place. Another disadvantage of the copy-back scheme for a GaAs microprocessor system is that it requires additional logic to maintain the dirty bits and to initiate the fetch-on-write and copy-back-on-replacement.

Multi-distance memory. When dealing with a new technology, such as GaAs, unconventional memory design techniques deserve greater levels of attention than they would normally receive. One idea that has found little use in the silicon technology environment and that may be appropriate for a GaAs system is a *multi-distance memory* (that is to say, a memory system in which the access time for different portions of the main memory is different). A multi-distance memory is beneficial because it improves the program execution time without run-time decision making (and consequently, without incurring hardware overhead costs). The disadvantages of this approach are the loss of locality exploitation and the loss of flexibility in program and data placement within the memory system.

Storage overhead. An important design consideration for both RISC and GaAs system architects is the movement of run-time complexity to compile time.[1] In the case of multi-distance memory, instead of using chip resources for the implementation of caches, the task of placing frequently used information in the fastest memory can be moved to the compiler.

How much extra area is needed in a cache as compared with an ordinary memory or with a multi-distance memory? A clue to the answer can be obtained by looking at a silicon-based design: that of the Zilog Z80,000 32-bit microprocessor's on-chip cache. The overhead in a cache is on a per-block basis, that is, the extra circuitry needed for a cache is replicated once per block. In the Z80,000, each block holds 16 bytes of data, eight data-validation bits, four bits for least recently used (LRU) data, 28 tag bits, and one tag validation bit, giving a total of 169 bits.[8] A total of 128 of the 169, or about three fourths of the cache bits, are used for net storage. The remaining one fourth cache bits, for overhead, use up approximately one half of the cache area because the overhead bits are relatively irregular in design (in contrast to the highly regular structure of the net storage cells). Further, since GaAs technology has more stringent fan-out limitations than silicon technology has,[3] one can assume that the overhead bits will consume an even larger proportion of the area in the former than they do in the latter. Thus, the net result is that the area available for data storage in a GaAs cache is less than half of that available for storage in a regular (non-cache) GaAs memory. Consequently, if a multi-distance memory is used, and if its "shortest-distance" part is placed on the same package as the CPU chip, the total storage capacity of the on-package memory will be about twice the net storage capacity of an on-package cache.

Cost of compile-time control. The loss of temporal and spatial locality exploitation capability is a major disadvantage of multi-distance memories. However, the additional burden placed on the compiler also deserves attention.

One anomaly in a multi-distance memory is that the various data may not arrive at the processor in the order that would seem indicated by the order of fetch instructions. Consider the effect of two consecutive fetches to memories at different distances:

(inst. i): Load R1,distant_memory*
(inst. $i+1$): Load R2,close_memory*

Because the close memory takes less time to respond to the fetch request than the distant memory, it is possible that its data will arrive at the processor before the data from the distant memory. This is called the *sequencing problem.*

This problem can be solved in a manner similar to the approach taken by the delayed fetch in the Stanford MIPS architecture.[9] The *delayed fetch* reorganizes the code to ensure that such problems do not occur. The reorganization happens at compile time, which implies that it will work only if the system for distribution of code to the distant memory elements is a static one. This solution may not work directly in a dynamic allocation system because the location of code is not known at compile time. Using dynamic code allocation in a multi-distance memory requires that there be some hardware mechanism (tags, for example) to aid in ensuring the orderly arrival of data from the memory.

Pipelined memory. Pipelining is a very effective method for achieving dramatic performance increases for relatively small costs. The classic example of silicon pipeline technology consisting of instruction-fetch, decode, and execution stages is a good illustration.

When one observes a relative increase in the instruction-fetch time that results from technological reasons, one can remedy the problem by breaking the instruction-fetch stage into multiple substages.[3] It is very fortunate that the relative memory-access time for GaAs systems—which is longer than that for silicon systems—lends itself well to pipelining. The longer GaAs memory system delay does not result from the lower raw speed of the memory itself, but from longer relative propagation delays between the CPU and the memory. By proper physical positioning of intermediate latches, the propagation delays are easily pipelined.

Figure 3 provides an illustration of a pipelined memory system with three stages. In the first stage the processor transmits the address (and data if write) to a latch physically near the memory. In the second stage the memory is accessed and the data is stored in the latch physically close to the memory (if read). In the third stage (if read), the data is propagated back to the processor.

Pipelining is a technique that can be employed in both cache-based and multi-distance memories, and for both instructions and data. Because pipelined memory systems can completely match the effective processor-memory bandwidth with the processor's requirements, they are extremely promising for GaAs.

The disadvantage of pipelined memory systems is that the increased pipeline depth they cause places severe requirements on the optimizing compiler. The reason for this is that branch delays[9] are longer, and early compiler efforts to replace the no-op instructions in the fill-in slots are most successful for short branch delays.[9] The implications of GaAs technology on these aspects of compiler implementation will be discussed by Milutinović et al.[1]

Virtual memory. The virtual memory (VM) system design is not impacted too severely by the use of GaAs technology. In many cache-design issues, the design of a GaAs microprocessor virtual memory system parallels some silicon-based design. The problems related to GaAs virtual memory management are treated by Silbey.[7]

---

*"Inst." stands for "instruction."

Arithmetic coprocessors

In designing arithmetic coprocessors for a GaAs environment, as in designing the CPU for such an environment, one should follow the RISC design philosophy.

Because of the different transistor count limitations of GaAs and silicon, certain operations typically included in silicon coprocessors cannot be included directly in GaAs coprocessors, and these operations must be synthesized. This is done by finding a small set of minimally complex operations that are sufficient to synthesize complex operations easily, yet require a transistor count low enough for implementation on a GaAs coprocessor.

The IEEE floating-point standard presents some problems in the GaAs environment. These exist because the standard requires several special cases for representation of numbers. Since GaAs works best with the RISC philosophy of "provide primitives, not solutions," the special cases should be handled by software rather than hardware. The IEEE standard provides a good general guideline for software implementations. However, the standard was designed with silicon architectures in mind. Since the GaAs coprocessor will be a small-area chip, exact adherence to the IEEE standard will probably not be possible for GaAs coprocessor implementations.

The essence of this discussion is that in a GaAs environment, the transistor count limitations play a major role. For example, in Figure 4, we present the relationship between number of devices per chip and speed for two IEEE format double-precision floating point coprocessors, one GaAs and one silicon, both designed by RCA and tested with the 1750A mix.[10]

An important consideration for the implementation of an arithmetic coprocessor is the method of communication with the CPU. The following paragraphs examine several possible schemes for implementing coprocessors in a GaAs environment, to determine the advantages and disadvan-

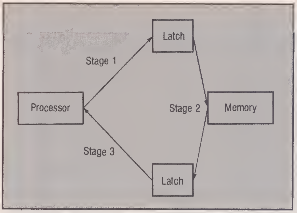


Figure 3. A three-stage pipelined memory.

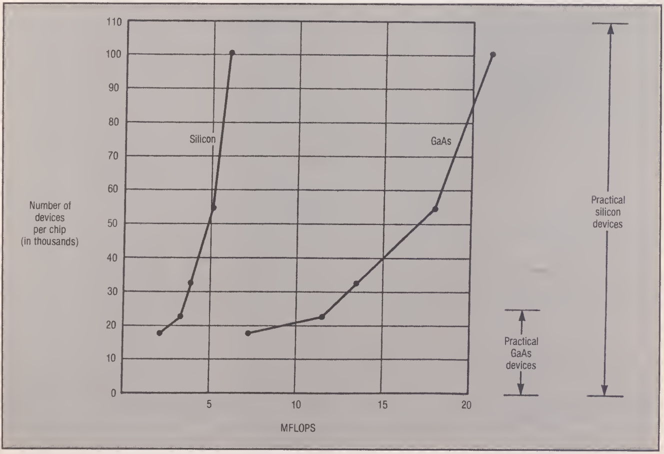


Figure 4. The relationship between coprocessor speed and device count for a silicon coprocessor and a GaAs coprocessor.

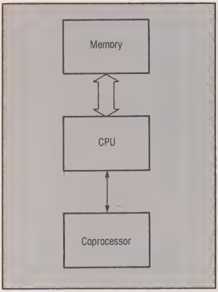


Figure 5. The coprocessor slave configuration.

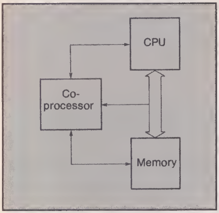


Figure 6. A set of possible coprocessor communication links.

tages of each. The two configurations discussed are a slave configuration and an expanded configuration.

Probably the simplest way of implementing a coprocessor in a system is to include it as a slave to the CPU. A block diagram illustrating this approach is shown in Figure 5. In this approach, the coprocessor is isolated from the rest of the system. All communication with the coprocessor is done through the CPU, and the coprocessor receives all instructions from the CPU. Unfortunately, there are several disadvantages to this approach; for example, because the coprocessor is a slave, the CPU must guide the coprocessor through each operation.

One way of reducing CPU involvement in coprocessor operation is to establish additional communication links—from the coprocessor to the rest of the system. The two most likely links are between the coprocessor and the system bus, and between the coprocessor and the memory. Figure 6 shows a set of possible coprocessor communication links.

We compare here the estimated design and performance parameters for RCA's E/D MESFET GaAs and CMOS silicon coprocessors.[10] In the first case, the number of devices is between 20K and 25K, and the number of floating-point registers is four. In the second case, the number of devices is between 60K and 75K, and the number of floating-point registers is 16. The expected performance ratio of the former to the latter is about 2:1 (11.6 MFLOPS versus 5.05 MFLOPS), for the 1750A mix. The expected cost ratio is about 100:1.

Input / output

The design of I/O systems for GaAs technology is not influenced by the change from silicon to GaAs as heavily as memory design. Nevertheless, GaAs technology does have a major impact on I/O design in the areas of I/O coprocessor design, DMA, and interrupt processing. The interested reader is referred to Bettinger.[11]

The concepts behind GaAs I/O coprocessor design are similar to those discussed for arithmetic coprocessors by Helbig and Milutinović,[12] since some of the interprocessor communication issues are similar. One design possibility is to move more "intelligence" into the I/O coprocessors. First and foremost among the benefits of intelligent I/O is the reduced amount of work assigned to the main processor. Giving intelligence to the I/O coprocessors minimizes the amount of interchip communication, which is important for GaAs.

The relative latency changes in the new technology, partly because applications and requirements may not have changed. Because of this, one can assume that it is safe to check for interrupts at the same time intervals as in silicon designs. During those same intervals, more instructions may be executed, which points out the possibility of interrupting the CPU on time boundaries other than instruction boundaries. This principle can be modified slightly to allow interrupts only on some common instructions such as load, store, or move. The compiler can then make sure that no sequence of instructions will leave an excessive gap in which interrupts cannot be recognized.

Because pipelined memories will very likely be used in GaAs systems, there are very few cycles for the DMA to steal. Some of those few might occur during cache misses, or during the no-op instructions, which are inserted concurrently with off-chip communications. Also, due to increased relative latencies, cycle-stealing DMA may become very difficult to implement.

Packaging technology for GaAs

Packaging considerations take on great importance in the GaAs environment.

Balance between processing, memory, and packaging technologies. Traditionally, in the context of a particular implementation environment, computer designers have striven to achieve a balance between processing capability and memory capability. A system with such a "matched" design provides the maximum performance for a given set of cost constraints. When rapid advances in processing power disrupted the balance between processing and memory in the 1960's, implementation strategies were invented in an attempt to regain this lost balance. As a result, various means, (caches, for example) have become widely utilized to enhance the capabilities of memory systems.

In the present-day implementation environment for silicon technology, on-chip logic and wiring delays significantly exceed off-chip delays. In this environment, packaging considerations are relatively unimportant and, as a result, packaging innovations to reduce off-chip delays have been less stressed than processing and memory improvements. In the GaAs environment, however, packaging design has become a critical bottleneck. Interchip signal delays are significant, and design techniques must be used to compensate for this undesirable characteristic.

Research in packaging materials and manufacturing techniques may once again allow computer designers to treat interchip communication as an ignorable problem.

However, as progress is made in packaging techniques, equally significant progress is being made in IC technology.

GaAs design methodology now consists of balancing the capabilities of three resources. Since advances in both processing and memory power are derived from IC technology advances, one expects only moderate difficulty in maintaining a balance between these two.

However, packaging advances are obtained independently of IC technology advances. Therefore, one can expect more difficulty in maintaining the constant balance of processing and memory with packaging, which is necessary if one is to be able to ignore interchip communication costs. To achieve balance between all three areas, there must be an increased emphasis on packaging technology improvements.

Issues in packaging. In spite of advances in packaging and interconnecting, the high switching speeds in GaAs force designers to view off-chip signal lines as transmission lines. This requires either that all signal lines be impedance matched, or that reflections be tolerated on the signal lines. Such matching of signal line impedances on a line that has multiple sources and sinks is quite difficult. This problem is of considerable importance when one is designing the memory system in GaAs computers.

The maximum speed of signals travelling between two chips is limited to the speed of light, that is, 1.0167 ns per foot. This is the *maximum,* however, and in practice the signals are much slower because of two factors.

- First, the signal flow in a medium other than air is slowed down by the medium's dielectric constant.
- Second, the signal flow is slowed because of capacitance loading on the signal lines and because of the signal-driver internal resistance.

Interchip wiring is generally laid out as screened or etched lines on a non-conducting medium. Common examples are printed wire boards on glass epoxy, and thick or thin film conductors on ceramic boards. The velocity of signal propagation relative to the speed of light is inversely proportional to the square root of the dielectric constant of the board material. Consequently, propagation delays of two inches per nanosecond are not uncommon.

In this interchip wiring environment, which is critical for GaAs, it is convenient to use transmission-line-type signal paths. For this reason, microstrip transmission lines are used, since they work well as surface lines on a medium with a ground plane. The microstrip lines are normally designed to have a constant, characteristic impedance, which is a function of the ratio of line width to dielectric thickness. The delay through these lines varies because of the impedance and, therefore, because of the "effective" dielectric constant variation caused by line width. Figure 7 plots the signal delay (in nanoseconds/ inch versus dielectric constant) for a family of line impedances that start at zero ohms (at which point the line width to thickness ratio—w/h—equals infinity) with no fringing and go to a maximum impedance (w/h going to zero) without a ground plane. Since the zero-ohm case is devoid of any fringing field, the effective dielectric constant is equal to the material dielectric constant. Therefore, the upper curve is also the delay curve for stripline transmission lines (line impedance is not a factor), in which the line is completely buried in the media between two ground planes. Conversely, the lower curve, which represents 0 ground, is applicable to coplanar lines. Microstrip lines much over 100 ohms are not realizable because of the narrow line widths required. Figure 7 also shows the common materials used and the range of their dielectric constants.

Two other factors affect the interconnection problem, and these result from

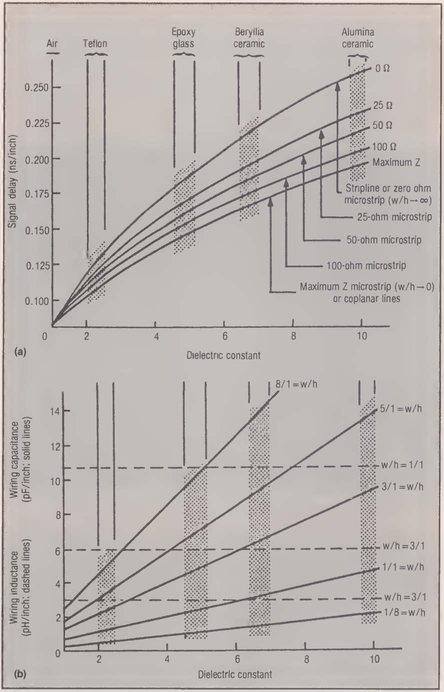


Figure 7. Wiring delay, capacitance, and inductance of various dielectrics. **(a)** Circuit delay versus dielectric constant (E_r) for several microstripline impedances (variable widths), including fringing and stripline (0 Ω microstrip has no fringing). **(b)** Capacitance and inductance versus E_r for several line width/thickness ratios (w/h).

having a data width of a large number of bits.

The first is the length of the lines to be used to make the interconnections. With, for example, 32 lines to go from one chip to another, even if the line impedance and delay can be controlled (or matched) for all lines, the length of the lines must be the same for all lines or bit skew will result. The solution is to either make all lines the same length (or close to the same length) or to build the electronics to de-skew the bits at the receiving end. (Actually, an approach that incorporates a little bit of both options is the best.)

The second factor is the physical difficulty associated with making the lines equal lengths. This must be done without causing the undesirable effects that result from signal reflection, but it must be done without taking up an excessive amount of board (package) space. (An example of the types of lines needed and the layout techniques required is shown in the picture of the GaAs chip in Figure 8. (Note the curves and the mirrored corners used to prevent reflections. And note the "waves" in the lines that are used to make the lines the same length.) If one uses too much space to make the lines the same length, then the board gets too big to manufacture, or the number of parts per board gets too low to result in an efficient design.

What is likely to happen is that, in the attempt to make large groups of lines equal in length, the package will be made of several layers. This will make the manufacturing easier and the control of characteristic parameters of the interconnects easier. However, multilayer packaging has an added difficulty: Feedthroughs, which are normally a 90 degree turn in the line, have to be made without reflection points (that is, they have to have 90-degree turns without mirror edges).

Another factor to consider is that buses are very difficult to implement. Having a source in the center of a matched impedance line means that both ends of the line must be terminated—making, for example, the effective impedance 25 ohms for a driver driving a line with a characteristic impedance of 50 ohms (and thereby requiring larger drivers). Second, having multiple loads on the line makes it difficult to control the impedance because the locations of the loads influence the ef-

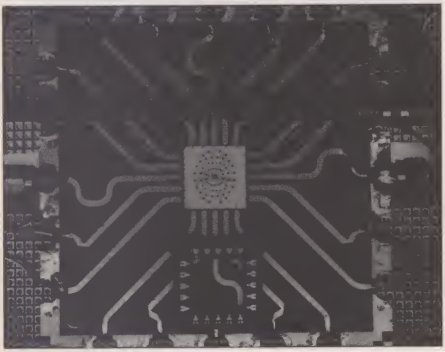


Figure 8. A packaged GaAs chip. (Photo used by permission of TriQuint.)

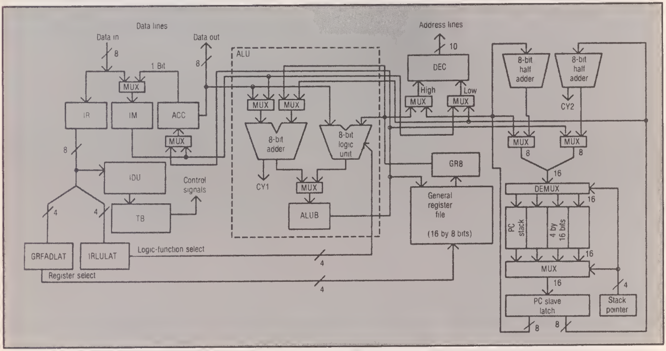


Figure 9. A block diagram of RCA's 8-bit GaAs Technology Demonstration Microprocessor system. ("ALUB," "GRB," "GRFADLAT," "IR," and "IRLULAT" are latches. "ACC" stands for accumulator; "CY1/2" stands for cycle #1/2. The "DEC" is a small combinational decoder and latch. "IDU" stands for instruction decoding unit. "IM" stands for immediate field buffer. The "TB" is the timing block. It includes an output latch.)

fective impedance. Third, the more one tries to simplify the operation by incorporating a bus, the harder it is to make the lines equal in length and not to have reflection points.

Examples of GaAs systems

An 8-bit GaAs demonstration system. The 8-bit GaAs Technology Demonstration Microprocessor (GTDM) is a pipelined, RISC machine.[4] It was originally targeted for 200-MHz operation and 100-MIPS performance. (Subsequently, it was redesigned for 200-MIPS performance.) A pioneering effort in GaAs technology, it has provided a vehicle for the development of a GaAs design methodology. A version of the GTDM consisting of the CPU, 512 words of memory, and some I/O capability is being constructed with leadless ceramic chip carriers. A block diagram of the microprocessor is shown in Figure 9. The VLSI layout, which was generated through extensive use of CAD techniques, is shown in Figure 10. A block diagram of the system is shown in Figure 11. The physical placement of chips on the board is shown in Figure 12.

GaAs fabrication in general is developing rapidly and is currently capable of integration of 1000 to 4000 gates with yields of 10 percent to 20 percent. A 1-micron double-level-metal E/D MESFET process is used in the GTDM. A single power supply is required, and the 0.2- and 0.8-volt levels represent logic zero and logic one, respectively. The design is primarily standard cell, with two handcrafted or custom-designed macrocells. These two macrocells are a 16-bit by 8-bit general register file, and a 4-bit by 16-bit program counter stack. Typical gates in the experimental GaAs E/D standard-cell family have delays between 100 and 500 picoseconds, and a maximum fan-out of four. The processor chip is 185 by 167 mils, and it dissipates about 858 milliwatts. It is packaged in a 48-pin leadless ceramic chip carrier.

The processor's instruction set consists of 23 instructions, 19 of which are 8 bits long and four of which are 16 bits long. The 100-MIPS goal required that an instruction be fetched every 10 nanoseconds. This processing rate is achieved through the use of simple, fast-executing, hardware-frugal instructions, and a pipeline that permits the simultaneous processing of several instructions.

The system includes four 256-by-4-bit Gigabit Logic 3-nanosecond GaAs memories (providing 512 8-bit bytes), 12 ECL input/output chips, and the 8-bit

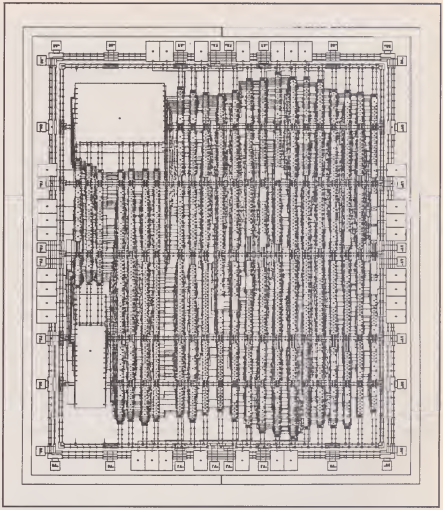


Figure 10. The VLSI layout of RCA's GaAs Technology Demonstration Microprocessor. (Photo courtesy of RCA Advanced Technology Laboratories.)

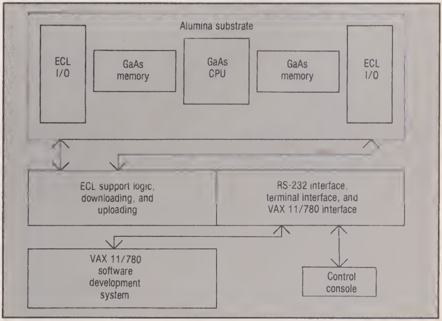

Alumina substrate
ECL I/O
GaAs memory
GaAs CPU
GaAs memory
ECL I/O
ECL support logic, downloading, and uploading
RS-232 interface, terminal interface, and VAX 11/780 interface
VAX 11/780 software development system
Control console

Figure 11. A block diagram of the GaAs Technology Demonstration Microprocessor system.

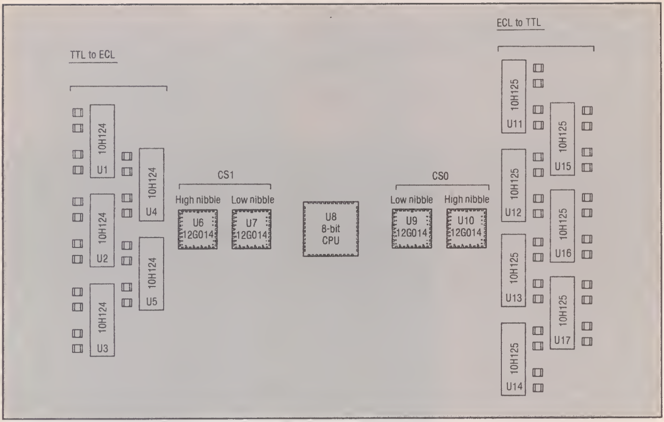


Figure 12. Physical placement of chips on the RCA GaAs Technology Demonstration Microprocessor board.

processor. A 16-bit address width allows for memory expansion to 64K bytes.

The 512 memory locations are high-order interleaved to provide the option of separate program and data storage in memory. If a program requires less than 256 bytes of memory, then it can be stored in one bank while data is stored in another. This arrangement allows two experiments to be run on the system. For programs requiring less than 256 bytes, no memory part is accessed more often than every 10 nanoseconds. Otherwise, if a program requires more than 256 bytes, it can request data from the same memory chip in the form of current instructions. This causes memory chip accesses to be 5 nanoseconds apart. If the system does not permit access at 5-ns intervals, then the clock frequency is lowered until the system works, thereby testing the cycle time of the memory parts as they function in the system.

System-level packaging of the microprocessor, GaAs memory chips, and ECL/TTL translators is done on an alumina substrate. Typical signal propagation delay time for transmission line interconnects built on alumina substrates is 250 picoseconds/inch. This translates into a delay time per inch that is 5 percent of the microprocessor cycle time.

A 32-bit GaAs system. RCA is also currently developing a 32-bit GaAs computer

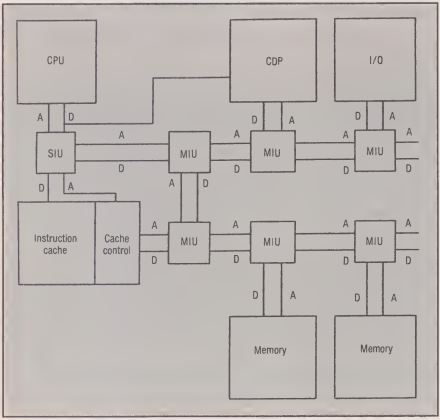


Figure 13. A block diagram of RCA's 32-bit GaAs microprocessor. ("A" stands for address bus; "D," for data bus; "COP," for coprocessor; "CPU," for central processing unit; "I/O," for input/output controller; "MIU," for memory interface unit; and "SIU," for system interface unit.)

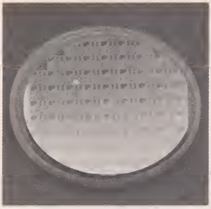


Figure 14. A GaAs E/D-MESFET test wafer. (Photo courtesy of RCA Advanced Technology Laboratories.)

system based on a single-chip GaAs processor. A block diagram of the system is shown in Figure 13. As shown in this block diagram, the system interface unit (SIU) multiplexes the two data streams that it receives (from a memory interface unit, or MIU, and the instruction cache) to send them to the CPU. The MIUs provide the necessary drive- and signal-conditioning functions to communicate from board to board.

A GaAs E/D MESFET demonstration wafer that has digital logic (1-micron two-level metal) related to that of the 32-bit GaAs microprocessor is shown in Figure 14. This wafer contains a 32-bit adder, a 16-word by 32-bit register file, a control PLA, and some process-monitor logic.

The processor design is based on the RISC design philosophy. A block diagram of the CPU is given in Figure 15. This allows for simple instruction decoding, which results in low resource requirements for decoding/control logic. The execution units (ALU, register file, and so on) also employ simple designs for low resource requirements. Details of this architecture can be found in Helbig and Milutinović.[12]

In the processor design, much emphasis was placed on maximizing the size of on-chip storage to reduce the need for off-chip communication. As a result, the resources saved elsewhere were allocated to the relatively large on-chip register file. The system components shown in Figure 13 are all part of a large pipeline. In the 32-bit system, the MIUs are junction points. Therefore, the delay between the processor and the closest MIU, the delay between MIUs, and the delay between MIUs and associated peripherals (coprocessor, memory, and so on) are all multiples of the processor cycle time.

At present, both computer-development projects are in progress under RCA funding. The 8-bit computer logic redesign for 200-MIPS performance has been finished. Standard cells to be used in its implementation have been redesigned several times and are now ready for the implementation stage of the project. Layouts of the chip and the hybrid on which it is to be mounted are underway. Processing of the chip and testing of the system are expected to be completed in early 1987.

The 32-bit computer's register stacks and adder have been laid out on the test

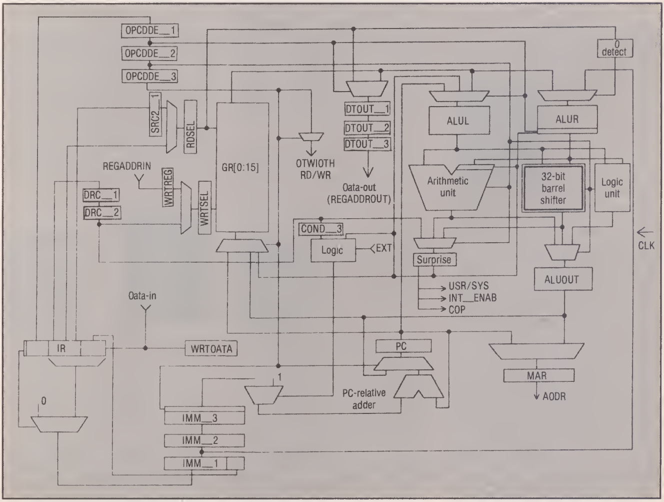


Figure 15. A block diagram of an experimental 32-bit GaAs CPU.

chip shown in Figure 14. Testing of this test chip is underway to verify design correctness and circuit performance.

Because of the high cost and low yield of GaAs wafers produced today, only a limited number of these test chips are available for testing. Even so, the register stack design has been proven correct and 5-ns access times have been measured. (The access time needed for 200 MIPS operation is 2.5 ns. The difference between the design goal and the measured performance is due to the measurement process. When used within the confines of a CPU, the register stack should provide the proper speed because it will not have to drive output lines with measurement equipment hanging on them.)

The development of the hybrid for the 8-bit machine has verified all expectations and fears. A delay to get the set of signals for memory referencing off of the CPU, to the memory, and back without detrimental skew has been accomplished. However, the design process required extra care and the total operation delay was, as expected, greater than one machine cycle time even for this machine's small memory capacity.

While this article discusses only a few of the system-level considerations for GaAs VLSI computer design and assembly, the critical ones have been identified. Unfortunately, most of the items discussed have been verified by the example systems under construction. ("Unfortunately" because they have verified that many more factors must be considered as architecture drivers than it was ever necessary to consider for silicon VLSI systems.)

Designing computers that incorporate GaAs VLSI will require greater effort than the corresponding effort required for silicon VLSI design. GaAs chips will be more expensive than silicon chips, and GaAs computer assembly will be more expensive than silicon computer assembly. The corresponding support software (compilers, assemblers, and code reorganizers) will have a lot more work to perform on the application programs, making code production slower and more expensive. However, reaching the goal of an increase over silicon system throughput of perhaps five times will be worth it.

The next things that must be tackled are the design of the interim system and compatible I/O systems. Interrupt systems are difficult to design in the GaAs environment because such an environment demands that instructions, data, and control-signal flow be of a pipelined nature and because of the severe limitation in the number of logic elements that can be put on any one chip. (One design has been conceived that will work; however, the response time to external interrupts is relatively slow.) GaAs I/O systems are difficult to design because of the increased ratio (over silicon) of CPU throughput to I/O channel throughput and because of difficulty involved with building a bus with GaAs circuits.

Considerably more work is needed to learn how to design and build GaAs VLSI computers. With the high payoff, though, the price will likely be paid. ☐

Acknowledgments

The authors appreciate the assistance they received from Wayne Moyers of RCA, and Chi-Hung Chi of Purdue University.

This research was sponsored by RCA.

References

1. V. Milutinović et al., "Compiler/Architecture Synergism in GaAs Computer Systems," to be published in 1987.
2. J. Hennessy and T. Gross, "Postpass Code Optimization of Pipeline Constraints," *ACM Trans. Programming Languages and Systems,* Vol. 5, No. 3, July 1983, pp. 422-448.
3. V. Milutinović, D. Fura, and W. Helbig, "An Introduction to GaAs Microprocessor Architecture for VLSI," *Computer,* Vol. 19, No. 3, March 1986, pp. 30-42.
4. W. A. Helbig, R. H. Schellack, and R. M. Zieger, "The Design and Construction of a GaAs Technology Demonstration Microprocessor," *Proc. Midcon 85,* Session 23, Chicago, Ill., Sept. 1985, pp. 1-6.
5. M. Horowitz and P. Chow, "MIPS-X," *Proc. Wescon 1985,* San Francisco, Calif., Nov. 1985, pp. 6/1/1-6/1/6.
6. J. E. Smith and J. R. Goodman, "A Study of Instruction Cache Organizations and Replacement Policies," *Proc. 10th Annual Int'l Symp. Computer Architecture,* Stockholm, Sweden, June 1983.
7. A. Silbey, "A Study of Cache Design and Virtual Memory Management in the GaAs Environment," Purdue/RCA Internal Report No. 14, May 1985.
8. D. Alpert, "Performance Trade-offs for Microprocessor Cache Memories," Stanford University Technical Report 83-239, Dec. 1983.
9. T. Gross, "Code Optimization of Pipeline Constraints," Stanford University Technical Report 83-255, Dec. 1983.
10. W. Helbig, "RISC vs. CISC, GaAs vs. Silicon, and Hardware vs. Software," *IEEE-CS Joint MTT/ED Meeting,* Princeton, N.J., Jan. 23, 1986.
11. M. Bettinger, "A Study of I/O Design Issues in the GaAs Environment," Purdue/RCA Internal Report No. 17, May 1985.
12. W. Helbig and V. Milutinović, "A 32-Bit GaAs Microprocessor," in *High-Level Language Computer Architecture,* V. Milutinović, ed., Computer Science Press, New York, 1986.

V. M. Milutinović's biography and photo appear following the Guest Editor's Introduction.

Alexander Silbey is a member of the technical staff of Gould, Inc.'s Computer Systems Division. He has an MS degree in electrical engineering and two BS degrees in engineering, all from Purdue University. He has published papers on high-level language computer architecture and advanced microprocessors, as well as on computer architecture for GaAs. His research interests include high-level language computer architectures for GaAs and parallel architectures for numeric processing. He developed cache and virtual memory approaches for a GaAs microprocessor as part of his research at Purdue University.

David Fura is a doctoral student in the Dept. of Electrical Engineering and Computer Science at Purdue University. His current research activity is in high-performance and highly reliable computer-system design. He received BSE degrees in electrical engineering and computer engineering from the University of Michigan in 1981, and the MS degree in electrical engineering from Purdue University in 1985. His master's thesis research consisted of processor architecture studies for the industrial implementation of a GaAs computer system. Before entering Purdue, Fura spent three years as a design engineer at Texas Instruments, Inc.

Kevin Keirn received his BS degree in electrical engineering in 1984 and his MS degree in electrical engineering in 1985 from Purdue University. He is currently a member of the technical staff of the Manufacturing Test Division of Hewlett-Packard Co. in Loveland, Colo. Prior to joining Hewlett-Packard, he was employed with Delco Electronics, Kokomo, Ind. His current research interests are programming languages, artificial intelligence, and computer architecture.

Mark Bettinger is a doctoral student in the Dept. of Electrical Engineering of Purdue University. His current research activity is in concurrent simulation of digital networks. He received the BS degree in electrical engineering from Purdue University in 1984 and the MS degree in electrical engineering from Purdue University in 1985. His master's thesis research consisted of system architecture studies for the industrial implementation of a GaAs computer system. While completing his BSEE he worked as a cooperative education student at the Federal Systems Division of IBM.

Walter Helbig is the unit manager for advanced information-processing techniques development in the Advanced Technology Laboratories of RCA Corp. He has been involved in all aspects of computer system development: working with both hardware and software, developing and applying both computers and peripheral equipment, and developing and applying both application and operating-system software for real-time applications. Helbig was one of the principal designers of the RCA 4100 series machine (a RISC type machine). He is presently technical director for both hardware and software designs for a reduced instruction set computer to be built in GaAs. Helbig received the BS degree in electrical engineering from the Milwaukee School of Engineering in 1952. He has published seven papers and has 13 patents.

William Heagerty is the unit manager of the Technology Development Laboratory in the Advanced Technology Laboratory of RCA Corp. He was program manager on the Purnell-II contract, which involved the manufacture of two VLSI microprocessors, an I/O controller, and program software. He was involved in the design of CMOS/SOS LSI and linear circuitry with emphasis on radiation-hardened applications. Heagerty is presently responsible for the introduction and application of new IC technologies. He has directed RCA's GaAs I R&D program entitled "Gigabit Logic Technology" for the past two years. He received the BS degree from Syracuse University in 1960 and the MS degree from the University of Pennsylvania in 1967, both in electrical engineering. He has published seven papers and has four patents.

Richard Zieger is a member of the technical staff in the Advanced Technology Laboratory of RCA Corp. His past assignments have included system architecture simulation studies and the design of a 3-nanosecond 32-bit GaAs adder. He is currently a principal designer of an 8-bit pipelined GaAs processor. Zieger received the BS degree in computer engineering from Lehigh University in 1984, and is presently a graduate student in the Electrical Engineering Dept. of Villanova University.

Bob Schellack is a senior member of the engineering staff at RCA Advanced Technology Laboratories. He is the project engineer of GaAs technology development for use in digital integrated circuit designs. His recent work has been involved in the design and fabrication of GaAs Manchester Encoder circuits and GaAs microprocessors. In addition, he has been the principal investigator for RCA's internal research and development of GaAs technology for the past three years. Schellack received his BS in electrical engineering from Drexel University in 1980 and his MS from Drexel University in 1983.

Walt Curtice joined RCA Laboratories, Princeton, N.J., in 1973 as a member of the technical staff in the Microwave Technology Center. Initially, he directed the development of second-harmonic-extraction TRAPATT amplifiers for X-band operation. He later developed the two-dimensional electron-temperature model of GaAs field-effect transistors and an improved MESFET model for GaAs integrated-circuit simulation; more recently, he has been directing the nonlinear-device modeling effort. In 1984 he received an RCA Laboratories Outstanding Achievement Award for the development of advanced techniques for the computer simulation of III-V compound FETs. Curtice has published more than 40 technical papers and has eight US patents. He received the BEE, MS, and PhD degrees from Cornell University in 1958, 1960, and 1962, respectively.

Readers may write to Veljko Milutinović at the Purdue University School of Electrical Engineering, West Lafayette, IN 47907.

The 5th International Conference On Entity - Relationship Approach

November 17 - 19, 1986 Dijon, France (Capital of Burgundy)

Major Theme: Ten years of experience in ER modeling

ER Conferences are intended to identify and encourage research, development, and applications of database and information systems based on the use of the entity-relationship approach. Ten years after the original paper in the ACM/TODS first issue, this conference wishes to offer a checkpoint on the usability of the ER approach for the design process and on its use as an operational tool, as well an insight into the theory of the ER model and into development perspectives.

Conference Organizers:

Conference Chairman: F. Bodart (Belgium), Program Committee Chairman: S. Spaccapietra (France), Tutorial Chairman: A. Flory (France), Organizing Committee Chairman: Y. Tabourier (France), American Coordinator: A.K. Arora (USA), Steering Committee Vicechairman: Jane Liu (USA).

Program Committee:

A.K. Arora (USA), C. Batini (Italy), M. Bouzeghoub (France), J. Bubenko (Sweden), A. Buchmann (Mexico), A.F. Cardenas (USA), S.M. Deen (UK), R.A. Elmasri (USA), J.P. Fry (USA), A. Furtado (Brazil), I.T. Hawryazkiewyiz (Australia), H. Kangassalo (Finland), L. Hazelton (USA), P.J.H. King (UK), I. Kobayashi (Japan), M. Leonard (Switzerland), T.W. Ling (Singapore), F. Lochovsky (Canada), S. March (USA), B. Moulin (Canada), E. Neuhold (Austria), A. Olive (Spain), C. Parent (France), A. Pirotte (Beigium), C. Rolland (France), H. Sakai (Japan), H.J. Schek (West Germany), A. Sernadas (Portugal), A. Solvberg (Norway), J.F. Sowa (USA), K. Subieta (Poland), Y. Tabouier (France), H. Tardieu (France), R.P. van de Riet (Netherlands), H. Weber (West Germany)

TUTORIALS

TUTORIAL 1: **ER approach: the past ten years and the next decade**
by Peter P. Chen, LSU and M.I.T. (USA)

TUTORIAL 2: **The database design process**
by Peter J.H. King, Birkbeck College, University of London (UK)

TUTORIAL 3: *Part 1:* **MERISE: an information system design and development methodology** *by Arnold Rochfeld, Consultant, Paris (France)*
Part 2: **Contribution of the ER approach to object management in an information system design workbench**
by Hubert Tardieu, SEMA-METRA, Paris (France)

PRESENTATIONS

* Opening address: Francois Bodart, Conf. Chairman (Belgium)

* Invited Paper: **Data and knowledge system: objectives of the European ESPRIT projectL** Uwe L. Haass, (C.E.C.)

* Invited Paper: **Complex object modeling for multimedia database systems:** Michel Adiba, (France)

* **Entity from a systems point of view:** P. Lindgreen (Denmark)

* **SA-ER: a methodology that links structured analysis and entity-relationship modeling for database design:** John L. Carswell, Shamkant B. Navathe, (USA)

* **Extending a dynamic modeling method using data modeling capabilities: the cases of JSD:** M. Mees, F. Put (Belgium)

* **Complex entitles for engineering application:** Klaus Ditrich, Willi Gotthard, Peter P. Lockemann, (Germany)

* **Some practices with the entity-relationship model:** a software workshop in econometrics: Th. Collet-Petitjean, Ph. Delhaye, B. Geubelle, J.C. Jacquemin, M. Moreau, (Belgium)

* **Modeling and manipulating objects in geoscientific database:** U.W. Lipeck, Karl Neumann (Germany)

* Invited Paper: **Object-oriented database systems: a report from a recent workshop:** Klaus Bittrich (Germany)

* **A database designer's workbench:** D. Reiner, G. Brown, M. Friedell, J. Lehman, R. McKee, P. Rheingans, A Rosenthal (USA)

* **Acquisition and use of contextual knowledge in a form-driven database design methodology:** Michael V. Mannino, Joobin Choobineh, Jing Jany Hwang (USA)

* **ER-based access modeling:** Gary Schuldt (USA)

* **Translation of an extended entity-relationship model to the relational model with inclusions:** N. Azar, E. Pichat (France)

* **Design and implementation of an extended entity-relationship data base management system:** Marc Junet, (CH)

Sponsored by

afcet

IFIP
WG 2.6 (Databases)

ER
Institute

In cooperation with

THE COMPUTER SOCIETY of the IEEE

acm
Association for Computing Machinery

- * **VERS-A vector based entity relationship database management system:** Asuman Dogac, Filiz Eyupoglu (Turkey)
- * **The use of ER data models in capability schemas:** Karen Ryan, James A. Larson (USA)
- * **BIER-the Behaviour Integrated Entity-Relationship approach:** J. Eder, G. Kappel, A.M. Tjoa, R.R. Wagner (Austria)
- * **A conceptual modeling expert system:** M.L. Kersten, CWI, and H. Weigand, F. Dignum, J. Boom (Netherlands)
- * Invited Paper: **Knowledge and data:** a report from the IFIP 2.6 DS-2 conference: Robert Meersman, IFP WG2.6 Chairman (Belgium)
- * **Semantics of query languages for the entity-relationship model:** Kazimierz Subieta, Marek Missala (Poland)
- * **The lattice structure of entity set:** P. Chen (USA), Mingrui Li (PRC)
- * **Extending the entity-relationship approach for dynamic modeling purposes:** Giorgio Bruno, Antonio Elia (Italy)
- * **Applying entity-relationship concepts to executable specifications:** Suzanne Sluizer, Stanley Lee (USA)
- * **SCRABBLE: a logical database management system:** Andre Flory, Lyon University (France) and Sal March (USA)
- * **Entity subtypes and the semantics of relational database functions:** L.I. Brady (Australia)

HOW TO REACH DIJON

Lyon is the nearest airport. From the Lyon-Part-Dieu station, a train will take you to Dijon in about 1 h 40. From Paris, the French superfast train (TGV), *the fastest train in the world*, will take you from the "Gare de Lyon" station to Dijon in 1 h 40. There are 9 or 10 services a day, from 6.35 a.m. to 7.46 p.m. A reservation is required: you may contact the SNCF (French Railways) counter at the airport, or any railway station or travel agency in Paris.

HOTEL ACCOMMODATIONS

A number of rooms have been booked in various hotels in Dijon. For any enquiry, or payment please contact: the Office du Tourisme - Service des Congres - Place Darcy 21000 Dijon (tel: 80.43.56.05, telex 350912 ESSI DIJON).

PROCEEDINGS

Preprints will be edited by AFCET and distributed at the conference. The final proceedings will be published by North-Holland in 1987. Proceedings from previous ER conferences may be ordered from North-Holland (2nd - ISBN# 0-444-86747-3, 3rd - ISBN# 0-444-86777-5, and 4th ER conferences) or IEEE computer society (4th ER conference, ISBN# 0-8186-0645-2). The first conference proceedings are out of print. For those interested in obtaining photocopies of the first proceedings, please send 50 U.S. dollars (per copy) to the ER Institute, PO Box 25130, Baton Rouge, LA 70894, U.S.A.

DEMONSTRATIONS

During the conference, participants will have the opportunity to attend demonstrations of products by vendors and of prototypes by research laboratories. Vendors and research laboratories willing to offer a demonstration should contact, as soon as possible: Prof. Jean-Jacques Chabrier, Laboratoire d'Infor-matique Faculte des Sciences Mirande, Campus Montmuzard, 21000 Dijon, FRANCE. Tel: 80.66.64.13.

SOCIAL EVENTS

A Cocktail Party in a Historical Place

A **cocktail party** will be offered to the participants by the City of Dijon, on Monday, November 17 (7.15 p.m.). This reception is scheduled at the **City Hall,** located in the *historical Palace of the Burgundy Dukes.*

Banquet at a Chateau

The **conference banquet** is scheduled on Tuesday, November 18, to be held at the **"Chateau de Clos de Vougeot",** an impressive Chateau (12th - 14th century) wonderfully located in the *heart of the most prestigious Burgundy wineyards,* 15 km from Dijon. This will be a unique opportunity for a **dinner you will never forget.**

Wine Auction Tour

An optional tour will be organized on Sunday afternoon, November 16. This tour will take you through the wineyards to Beaune (35 km from Dijon), where you will be able to attend the **famous auction sale of the wines of the Hospices de Beaune.** The Hotel-Dieu (hospital), part of the Hospices, is a beautiful building in the typical Burgundy-flemish style of the 15th century (a visit is a must). Wineyards were donated to the Hospices through centuries; the wine is sold every year on the third Sunday in November. This is the **most important charity sale in the world.** A *wine-tasting festival* is also held on this special day.

REGISTRATION FEES

	before 10/27	after 10/27
Regular fees including AFCET Members	2900 FF	3600 FF
Speakers	2100 FF	2600 FF
Full-time students under 29	300 FF	400 FF

Regular and speakers rates include preprints, lunches, cocktail and the banquet. For French non-AFCET members, add a 80 FF, amount of AFCET membership for 1987. Student rate does not include meals.

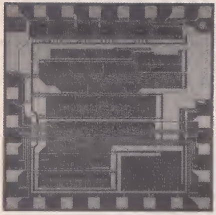
Pre-conference tour to Beaune (Sunday, November 16, 1986: 140 FF. Banquet for accompanying persond, per person: 410 FF. These amounts should be paid in advance, together with the registration fees.

FURTHER INFORMATION

A complete brochure with conference registration form and hotel reservation form may be obtained from:

(1) AFCET, 156 Boulevard Pereire, F 75017, PARIS FRANCE, Tel: (1) 47.66.24.19.

(2) Dr. Adarsh K. Arora, Gould Research Center, 40 Gould Center, Rolling Meadows, IL 60008, U.S.A. Tel: (312) 640-4712.



The MDC2901, a 4-bit GaAs microprocessor circuit that contains 1860 transistors. (Photo courtesy of McDonnell Douglas Microelectronics Center.)

A 32-bit RISC Implemented in Enhancement-Mode JFET GaAs

Terrence L. Rasset, Roger A. Niederland, John H. Lane,
and William A. Geideman
McDonnell Douglas Astronautics Company

This CPU architecture exemplifies the special design approaches being developed to take advantage of GaAs technology for high-speed processing.

Recently, considerable attention has been focused on gallium arsenide (GaAs) semiconductor process technologies for very high speed digital integrated circuits. This attention is motivated primarily by the ability of GaAs transistors to switch much faster with lower power consumption than transistors in silicon technologies.[1,2] Soon to be available with higher switching speed and more transistors than the fastest silicon technology (emitter-coupled logic), GaAs digital ICs will be a boon to developers of supercomputers and specialized high-speed microprocessors, such as digital signal processors. GaAs transistors are also much more resistant to temperature extremes and to ionizing radiation than silicon, important features for applications requiring operation in harsh environments.

This does not mean that GaAs will replace silicon; rather, where the higher cost of GaAs is warranted by application requirements that only GaAs can satisfy, it will augment silicon. The high cost of GaAs chips is due mainly to the scarcity of gallium and the inferior quality and difficulty in manufacturing the gallium arsenide compound. Unlike silicon, which is very uniform and pure, GaAs has many defects and its characteristics may vary considerably from ingot to ingot, affecting yield. Also, the wafers have a tendency to break during processing.

GaAs material defects and problems controlling noise margins across the chip affect processing yields and limit chip die sizes, thus limiting the number of possible transistors on a chip. Power consumption also limits the chip size. Although GaAs consumes much less power than silicon ECL, it usually consumes more than silicon MOS processes. As in silicon, power consumption is related to the switching speed of the transistors; generally, the faster the gates, the more power they consume.

As the quality of the GaAs material has improved and the processing technologies have matured, greater circuit densities have been achieved. During the last few years, a great deal of development work has centered on gate arrays and memory chips, resulting in significant advances. By 1983 experimental 1K static RAMs had been announced, and in 1985 4K and 16K static RAMs and 2000-gate arrays were announced.[3] Recent announcements have included experimental 5K-gate arrays.

To fully exploit GaAs in digital logic circuits requires special design approaches. Because gates switch so fast, the ratio of on-chip propagation delays to off-chip communication delays is greater than in silicon. Even in today's supercomputers, which use very high speed ECL logic, chip-to-chip communication is a more important factor in the computer's cycle time

0018-9162/86/1000-0060$01.00 © 1986 IEEE

than on-chip gate delays. This suggests that higher levels of integration and novel computer architectures are necessary before GaAs technology will pay real dividends in high-performance computers.

A number of efforts to develop GaAs computers are currently in progress. One of these projects is at McDonnell Douglas Astronautics Company, where a single-chip GaAs microprocessor and a single-chip floating-point coprocessor in a proprietary GaAs junction field effect transistor process are under development. The goal of this project is twofold: first, to develop and demonstrate GaAs JFET technology, and, second, to produce a microprocessor for very high data rate real-time processing.

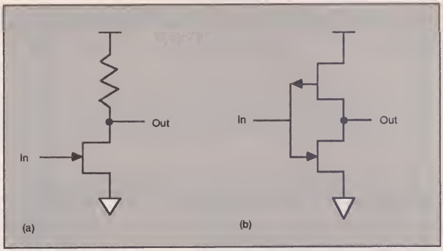


Figure 1. GaAs inverter circuits: resistive load inverter **(a)**, and complementary inverter **(b)**.

GaAs E-JFET process

The N-channel enhancement mode junction field effect transistor (E-JFET) is fabricated by ion implantation in a semi-insulating GaAs wafer.[4,5] The process steps are similar to silicon wafer processing except that no furnace diffusion steps are used and all etching is dry, using plasma and reactive ions. The JFET P-N junction is formed by implanting silicon ions to form the N-region and then implanting magnesium over the silicon-implanted region to form the P-region.

An added advantage of the JFET process is that the implantation steps can be reversed to produce a P-channel device. Thus, P-channel and N-channel transistors can be combined on the same wafer to produce complementary logic in GaAs.

The schematic for both the complementary and resistive load inverters is shown in Figure 1. Both of these circuits can be manufactured on the same wafer, giving the circuit designer the ability to save power where speed is not critical. These are simple, low-transistor-count devices with no level-translation required between any of the different circuit combinations, as is required with many other GaAs processes. Thus, they allow the use of direct coupled FET logic (DCFL), with a voltage swing of about 0.8 volts and a noise margin of 200 millivolts.[4]

The low power dissipation, simple transistor configurations, low transistor count per logic gate, and ease of processing make the GaAs E-JFET technology a good candidate for large-scale integrated circuits. This has been demonstrated with the successful design and fabrication of a 1K static RAM with a wafer yield of 47 percent, and a fully functional emulation of the Advanced Micro Devices AM2901 four-bit microprocessor slice. A 4K static RAM containing over 25,000 transistors on a 4.6 mm × 4.1 mm die is currently in processing, with functional parts expected by the end of 1986.

Even though the GaAs JFET technology is still maturing, it has proven viable for high-speed and low-power digital integrated circuit applications. The JFET process has performed well in adverse environmental conditions, not only outperforming all silicon technologies but also all other GaAs processes in resistance to radiation effects.[6]

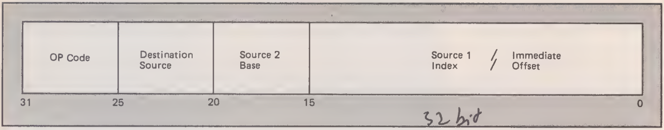
Currently, E-JFET logic gates are sensitive to circuit loading, which limits gate fan-outs, and to other types of gate loading. The loading on a particular net is composed mainly of interconnect resistance, capacitance between interconnects, interconnect crossovers, and gate capacitance of the transistors connected to the net. This sensitivity has a significant effect on the circuit designs developed and puts an extra burden on the architecture, logic, circuit, and layout designers, who must work as a close-knit team to produce desirable results. The loading effects will be reduced in the near future with self-aligned gate structures, which will reduce the sidewall capacitance of the gate. The effects of interconnect are being reduced by lowering the sheet resistance of the interconnect and by decreasing the dielectric constant of the insulating material between the interconnect layers to lower the interconnect capacitance.

The GaAs JFET sensitivity to loading also has a considerable effect on off-chip communication, in which it can adversely affect system performance. In GaAs, the ratio of on-chip delay to off-chip delay is much greater than in silicon technologies, thus making the system partitioning problem even more critical than in silicon,[3] where the slower gates and larger scale of integration lessen chip-to-chip communication effects. Therefore, the search for a suitable GaAs computer architecture is strongly influenced by the need to lessen interchip communication. This implies that large-scale integration capable of producing single-chip microprocessors and compact systems is even more important in GaAs than in silicon.

Architectural overview

In developing the GaAs microprocessor, we wanted an architecture that could be initially implemented in less than 25,000 transistors, yet would support the Ada programming language and be viable well into the next decade. Because the GaAs process is maturing and higher levels of integration will be available by the end of the decade, the architecture should be able to grow, to take advantage of these increased levels of integration. For efficient support of Ada, the architecture has to be a good compiler target and should provide a large address space. To provide sufficient precision for the planned applications and a large linear address space, we preferred a 32-bit architecture.

To achieve our rather ambitious architectural goals within such a small transistor budget, we followed the reduced instruction set computer philosophy in developing our architecture.[7,8] The RISC approach favors simplified instruction sets with few format options or addressing modes. It requires that the implementation cost of every instruction and hardware feature be justified in terms of frequency of use in the execution of programs, resulting in a streamlined, efficient instruction set architecture.[9] With a simple instruction set, the hardware can be optimized to execute the instructions that are used most frequently in the application programs. Other, more complex instructions are synthesized, as needed, out of the simpler instructions. In this way, for complex operations, such as a procedure call, we can generate the code



32 bit

Figure 2. Instruction word format.

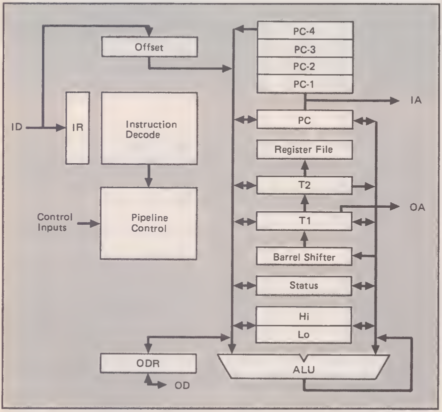


Figure 3. CPU block diagram.

optimally for each instance, rather than creating a large, complex instruction that attempts to deal with all cases (and probably satisfies few). The RISC approach results in a greatly simplified control structure (usually hardwired), which, in addition to being very fast, leaves more chip area available for performance enhancements to the data path.

At the time we started developing the architecture, the leading RISC projects were the University of California at Berkeley RISCs I and II,[10] the Stanford University MIPS,[8] and the IBM 801.[11] Of these, the Stanford MIPS project influenced our architecture the most. The MIPS microprocessor was implemented in an NMOS silicon semiconductor process using approximately 25K transistors. The architecture of the MIPS was compiler driven; that is, the instruction set was determined on the basis of its utility as the target for an optimizing compiler. The MIPS group felt that compiler optimization techniques have reached such a level of sophistication that computer architecture assumptions with regard to the execution of compiled code must be reexamined. They concluded that a simplified load-store architecture (i.e., only load and store instructions reference data memory; all others are register to register) with no hidden microarchitecture could deliver better performance in the execution of compiled code than could a complex instruction set computer (CISC).[12] The MIPS approach requires sophisticated register allocation and other optimization techniques in the compiler to achieve full performance. MIPS was not intended to be hand programmed at the machine instruction set level.

A unique feature of the MIPS microprocessor that was incorporated into our GaAs microprocessor is the use of pipelined instruction execution without hardware interlocks to detect resource conflicts. Instead, the code emitted by the compiler is rearranged to eliminate conflicts by a tool called a reorganizer. There are two main advantages to reorganizing at compile time rather than detecting at run time: first, the hardware necessary to control resource allocation and detect conflicts is eliminated, and, second, execution is faster because no instructions are suspended while waiting for an earlier instruction to finish using a resource (e.g., waiting for an instruction to write to a register).

In addition to rearranging code to eliminate pipeline conflicts, the reorganizer also attempts to replace NOPs (null operations) that result from branch delays and data load latencies with useful instructions. Delayed branches[13] are another feature of the RISC philosophy. Rather than fetch and discard the instructions immediately following a branch that is taken (as most CISCs do), most RISCs execute these instructions. At the worst, NOPs are inserted into these instruction slots, but with a one-instruction branch delay the MIPS reorganizer was able to move a useful instruction into the branch delay slots 90 percent of the time. Another way to look at this is that data taken from the MIPS shows that an average of 21 percent of the executed instructions were performed in the delay cycle following a taken branch.[8] If these instructions were fetched and not executed, 21 percent of the cycles would be wasted. Similar data was gathered for programs executing on a VAX 11/780.[14]

The architecture we developed is very simple and regular, yet has hardware resources in the data path that are not found on more complex microprocessors. All instructions are one word long, are available in either an immediate or register-to-register format as shown in Figure 2, have fixed fields, and execute in one cycle. Unlike most complex instruction set microprocessors, the critical delay path in this microprocessor is in the data path, not the control path. That is, the execution cycle time is completely determined by the time it takes to access the operands, perform an ALU operation, and save the result.

The microprocessor. The microprocessor architecture and implementation were driven by the constraints of the GaAs process. So that the microprocessor would be buildable in the near term, a strict transistor budget was established at less than 25,000 transistors.

The fixed format of the instruction leads to a very simplified instruction decode, reducing both the number of transistors needed for its implementation and increasing the speed of the decode. The instruction decode accounts for less than two percent of the total transistor budget and controls the functional blocks of the microprocessor with only 31 control lines. The only other significant control circuitry on the microprocessor is the pipestage control, leaving 95 percent of the transistor budget for implementing the data path.

The microprocessor's data path is partitioned into functional units as shown in Figure 3. These functional units are used to implement the instructions in a manner which gives the most performance for the transistors. The PC contains the current program counter plus the three most recent values for exception recovery. The register file contains 17 general-purpose registers and a constant zero generator. A 32-bit word can be shifted in either direction any number of bits in either a logical or arithmetic manner in the barrel shifter. The two registers in Hi and Lo can be used as general-purpose registers but they also contain multiply and divide support circuitry. The ALU performs all logical and arithmetic calculations. Offset provides alignment for immediate data. The operand data register (ODR) communicates with operand data memory. The temporary registers, T1 and T2, are used for pipestage alignment and are transparent to the user. A status register is contained in Status. All of these registers are mapped into the register file address field (except the temporary registers), as in Figure 4.

All instructions contain either three register operand addresses or two operand addresses and 16 bits of data. All register address fields are five bits long, allowing up to 32 registers in the architecture, of which 22 are implemented. Program-accessible registers are addressed by these five-bit fields in the instruction, providing a simple, uniform view of the register set.

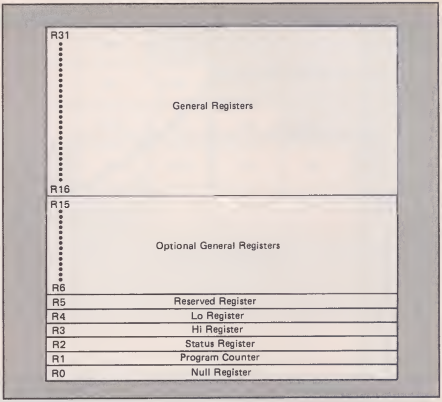


Figure 4. CPU register set.

Since the transistor budget would not allow the implementation of a full multiplier, a two-bit-per-cycle iterative Booth's algorithm was implemented. The multiplication control circuitry is incorporated in the Hi and Lo register pair and the ALU. This added a minimum of transistors and provided a means to accomplish a 32-by-32-bit multiply in 16 cycles. The Hi and Lo registers are also used to implement an iterative, nonrestoring divide algorithm. A 64-bit dividend can be divided by a 32-bit divisor, yielding a 32-bit quotient and a 32-bit remainder in 32 cycles.

All of the functional units are utilized during different instruction pipe stages. The pipe stages provide the flow necessary for overlapping instruction execution to increase resource utilization, resulting in superior performance.

Determining the number of pipeline stages. A difficult (and contentious) decision, intimately involved with the memory system design, was the choice of the number and type of execution pipeline stages. The decision centered on the problem of extending the execution pipeline into the memory and pipelining memory references. We quickly concluded that the four-stage pipeline discussed below was the ideal minimum. But increasing the number of stages to five or six, with the extra stages added to the memory access stages, would allow more time for memory accesses and therefore would make the memory system easier to build (or, some would argue, buildable at all). The six-stage pipeline would provide two stages for instruction fetch and two for operand read/write. This would require that the memory system be pipelined, the address being sent to the memory in the first stage and the data accessed and delivered to the microprocessor in the second. A new memory access would be achieved on every cycle by overlapping the address decoding with memory chip accessing and data delivery. This means that the memory word received by the microprocessor during cycle j would be the word read using the address sent out on cycle $j-1$. For the five-stage case, only the data memory accesses would be extended to two cycles. This is based on the fact that most instruction references are sequential and very localized, which means that an effective instruction cache memory is more realizable than an effective data cache.

At first the six-stage pipeline seemed to offer considerable relief from a very difficult problem. Unfortunately, nothing is given for free: The six-stage pipe increases branch delays from one cycle to two and load latencies to two cycles. Data gathered from benchmark programs taken from real-time system applications and applied to four-, five-, and six-stage pipelines showed that the reorganizer was not able to effectively move useful instructions into the extra delay slots. The performance loss

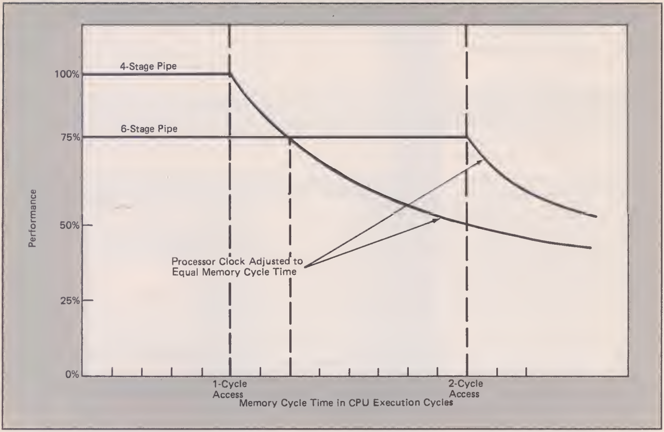


Figure 5. Performance model: 4-stage pipe vs. 6-stage pipe.

due to NOP insertion was from 20 to 30 percent for the six-stage pipeline and approximately half of that for the five-stage pipe relative to the four-stage pipe. The effect on microprocessor performance is illustrated by the graph in Figure 5, which shows microprocessor performance degradation for the four- and the six-stage pipelines as the memory cycle slows relative to the microprocessor cycle speed. The vertical axis is the net microprocessor throughput normalized to the four-stage implementation and the horizontal axis is the effective memory system access time. In reality, due to extra logic delays and problems inherent in clocking the memory pipe stages synchronously with the microprocessor clock, performance with six stages actually falls off before the memory access time reaches twice the microprocessor cycle time. Another problem with the six-stage pipeline is that it would require a more complex master pipeline control due to the extra states, more transistors for saving an additional copy of the PC, and another temporary register in the data path. The extra registers also add extra loads that must be driven by the data buses, which would have some effect, albeit small, on the execution cycle. With a four-stage pipeline, we can run the microprocessor's clock at least 25 percent slower than with a six-stage pipeline and still achieve the same net throughput.

The pipeline. The instructions can be divided into two general categories: computational and memory reference. The instructions finish their execution in four pipe stages though some instructions do not use all four. These stages are shown in Figure 6.

The instruction addressed by the program counter is accessed during the first pipe stage, called instruction fetch. The lower half of the fetched instruction may contain a data value; therefore it is loaded into the offset register concurrently with the instruction register loading.

The instruction register contents are used in the ALU pipe stage. The operand address fields of the instruction register require no decoding and are sent directly to the registers so the operands can be accessed immediately. The simple instruction decode allows the control lines to become valid before the access of the operand is complete; therefore an instruction decode pipe stage is not necessary. The operands are latched at the arithmetic unit and the barrel shifter, allowing both buses to be used during the second half of the pipe stage. The result is computed and transferred to T1, where it is loaded at the end of the ALU pipe stage. The results may also be loaded into the program counter, the Hi or Lo register, or the status register. If a store operation is in progress, the data is transferred to the operand data register (ODR) during the second half of the ALU pipe stage.

At the beginning of the third pipe stage, operand fetch, the contents of T1 will either contain the result of a computation or an operand address. In either case the contents are available on the operand address bus (OA). This provides not only the address for the operand memory, but a window into the microprocessor for testing purposes. The operand memory can be read or written to during this stage. If it is a read operation, the operand memory outputs the contents of the addressed location on the operand data bus, which is enabled onto the internal bus, making the data available to T2 during the second half of the pipe stage. T2 will be loaded with the operand memory contents during a load operation, displacing the address in T1. Otherwise, T1 is simply transferred to T2.

During the last instruction pipe stage, write back, T2 is transferred into the register file, if that was the destination.

The minimum microprocessor cycle time is the time it takes for the critical path

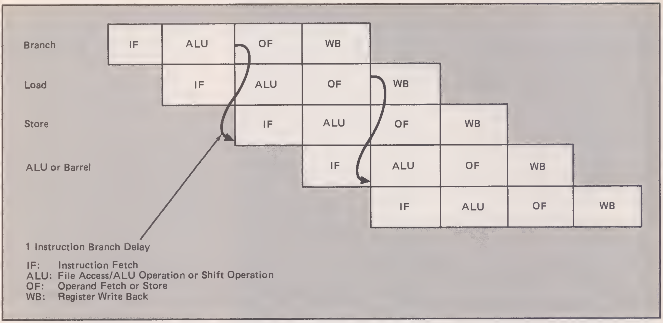


Figure 6. CPU pipeline flow.

in the ALU pipe stage, which starts with the instruction register outputting the operand addresses and ends with T1 being loaded with the results. Between these operations, the register file must be accessed, the operands transferred to the ALU, the add performed, and the results loaded into T1. Forty percent of the cycle is allocated for transferring data over the internal buses between the functional units; the rest of the cycle is split between the adder and the register file access.

Traditional microprocessor design uses buses for communication between different functional blocks, but, in order to get the most performance out of the GaAs circuits, interconnects must be kept short, with as few crossovers as possible.

Just as in silicon, precharging techniques are used to reduce the bus transfer time. The time available for precharging the buses is equal to the slowest functional unit. The functional units operate at much higher speeds than their equivalents in silicon, making the time available for precharging in GaAs significantly less than in silicon. Unfortunately, the bus loading is not significantly different. Also, clock skew must be controlled very closely to regulate this short precharge time. The bus design is a major factor in the critical path of the microprocessor and therefore received considerable attention.

A two-phase nonoverlapping clock scheme is used. This scheme maps well with the bus, giving two active pulses per pipe stage when the buses can be used for transferring data. The clocks have to be controlled so that the clock skew is kept below half a nanosecond between the registers. The loading and interconnect lengths must be tuned for each branch of the clock distribution tree. To reduce the effects of clock skew across a 32-bit data path, all registers are clocked from the least significant bit, giving the adder more time since its most significant bits are a function of the least significant bits.

Exceptions, memory wait states, and hold states are handled by the microprocessor's pipe stage controller, a simple state machine with eight states, the majority used for startup and wait states. When an exception is taken, the state machine is forced to the IF state. All of the information necessary for a complete recovery from any exception is obtainable from the program counter's history file and the status register. The history file contains the three previous values of the program counter. A startup sequence is necessary to start the pipe stages after an exception; this is provided by the states IF, ALU, OF, and Exe. Exe is the normal operating state. When the memory needs added time for access, the machine goes from Exe to the wait state. It will remain there inhibiting all pipe stages until the memory is ready; then on the next clock edge it returns to its normal state. Each of the states in the startup sequence has a wait state associated with it.

ALU implementation. A good example of the circuit design decisions that must be made in GaAs is the implementation of the microprocessor's arithmetic logic unit. Because the ALU is in the microprocessor's critical timing path, a good, high-speed design is essential. The ALU is decomposable into separate adder and logic sections, with the adder being by far the more complex. A high-speed adder design in GaAs should keep gate loading to a minimum, simultaneously keeping the number of gates in the longest path to a minimum. The adder must be 34 bits wide to implement the multiply algorithm. The four candidates considered for the adder design are listed in Table 1.

Table 1. Adder types.

	Transistors in Critical Path	Average Load	Maximum Load
Ripple-carry	82	1	2
Carry-select	37	1.9	18
Carry-lookahead	11	3.4	8
Modified* Brent & Kung	23	1.5	3

*Includes logical unit; all others exclude logical functions.

Many of the criteria followed in selecting an adder design for GaAs are the same as for silicon, but the emphasis is different. Due to crossover and communication effects, the layout is much more critical for performance in GaAs than in silicon. As in other integrated circuit technologies, an efficient use of transistors and a regular layout are preferred. In silicon, the efficient use of transistors makes economic sense; in GaAs it is absolutely essential. In silicon, gate delays are so dominant that other factors can be ignored and extra complexity can be introduced if it reduces the number of gate delays in the critical path. For implementation in the GaAs JFET process, we rejected the carry-lookahead adder because of the number of transistors required and the irregular layout. Delays attributed to crossovers were nearly equal to the delays due to fan-out.

The adder we chose is a modified version of an adder described by Brent and Kung.[15] This adder is characterized by simple cells with low gate fan-out and a very regular layout. The critical path contains 23 gates (compared ~~with 11 in the carry-lookahead adder) with an average~~ load of 1.5 gates. The ~~maximum loading on any one gate is~~ three gates. In addition, the loading from interconnects and crossovers is minimal since most of the connections are local.

Another advantage of the modified Brent-Kung adder is that its layout and control signals made it possible to superimpose the logical unit over the adder. This added only seven transistors and one resistor per bit and one gate delay to the critical path.

The ripple-carry adder has a very regular structure with exceptionally low fan-out, but with a 34-bit adder the number of gates in the critical path (82 gates) is unacceptable for a high-speed circuit. The carry-select adder, with 37 gate delays in the critical path, has a gate driving 18 loads, slowing the adder to an unacceptable speed.

The floating-point coprocessor

For efficient execution of floating-point arithmetic, an optional floating-point coprocessor chip is also being developed. The coprocessor is optimized to perform floating-point arithmetic on either single precision (32-bit) or double-precision (64-bit) data in the IEEE format. Due to the complexity of floating-point operations, the coprocessor does not follow the single-cycle instruction execution scheme of the microprocessor. Rather, each floating-point instruction is a complete floating-point operation, including programmable rounding and normalization.

The full IEEE standard is too complex to be implemented fully in hardware; therefore only the essential computational elements were included. To implement the full standard, software must be used to augment the core hardware resources. For example, arithmetic exception detection is fully implemented in hardware but exception handling is not. Whenever an exception, such as overflow or underflow, is detected by the coprocessor, a signal is sent to the microprocessor. A software handler executed by the microprocessor determines the cause by reading the coprocessor status register and performing the application-specific exception handling. This strategy results in a control area that consists of two relatively small (compared to other IEEE floating-point units) PLAs. The coprocessor does contain full hardware support for arithmetic operations, including eight double-precision operand registers and double-precision significand- and exponent-processing units.

The coprocessor depends on the microprocessor to perform all memory accesses. The coprocessor monitors the instruction stream, picking off the floating-point instructions as they are fetched by the microprocessor. During the execution of floating-point arithmetic operations, the coprocessor functions independently of the microprocessor. For floating-point load or store instructions, the microprocessor computes the memory address and initiates the memory accesses; then it allows the coprocessor to load data from or enable data onto the operand bus. For conditional branches on floating-point conditions, the coprocessor provides a True/False-valued signal to the microprocessor. The microprocessor computes the branch target address and loads this into the PC if the signal from the coprocessor is True.

To reduce the demands placed on the microprocessor for instruction fetching and to allow greater concurrency between the microprocessor and the coprocessor, two coprocessor arithmetic instructions are packed into a 32-bit instruction word. To further support parallel, concurrent operations, two floating-point coprocessors (plus two other coprocessors) may be used with a single microprocessor. This feature was added to the interface design primarily to support multiply-intensive applications. Because a full multiplier would not fit on the coprocessor chip, the multiply instruction uses an iterative, two-bit per cycle algorithm. The result is multiply execution times three to four times slower than add instruction execution times. The reorganizer assists the compiler in packing floating-point instructions and in scheduling two floating-point and one fixed-point computational streams.

Coprocessor implementation. Figure 7 is a block diagram of the main functional units in the floating-point coprocessor. Independent, dedicated processing elements are used wherever possible to maximize the parallelism of operations within the coprocessor chip. The coprocessor is partitioned into two semi-independent sections: the bus interface section and the arithmetic processing section. In the arithmetic processing section, the exponent processor performs all operations involving the exponents while the significand processor performs arithmetic operations on the significands. Some hardware components were added to make the execution time of all instructions fixed, with no operand dependencies, to facilitate instruction scheduling by the reorganizer. Exclusive of peripheral circuits, approximately 80 percent of the coprocessor chip transistors are dedicated to computational elements.

The same implementation considerations hold for the coprocessor as were discussed for the microprocessor. However, due to the nature of the operations performed by the coprocessor, data busing and control signal distribution are much simpler in the coprocessor. Each floating-point arithmetic operation follows a fixed sequence of operations that requires multiple processing cycles per instruction, unlike the microprocessor, which must execute a new instruction on every cycle with a new set of operands. The coprocessor reads a set of operands from the register file at the start of an instruction execution, and then, at the end of several execution steps, the result is written into the destination register. This reduces the requirements on the internal operand data buses and allows the control signals to be decoded in advance and routed to the data path components without strict time requirements. Thus, there is no need to allow any control decode time in the length of a processing cycle, and the cycle time is based entirely on the data path delays.

The floating-point coprocessor contains in a status register condition codes that are set as a result of arithmetic operations. The bus interface section uses these flags to execute conditional branch instructions concurrent with arithmetic instruction execution in the arithmetic processing section. This allows the microprocessor to branch on coprocessor busy or on the results of the first of the two arithmetic instructions while the second is being executed. In this way, instruction fetch delays that would otherwise result from a branch to a new block of code can be avoided.

The coprocessor uses the same clock as the microprocessor. This is necessitated by the close cooperation with which the two processors must operate. In particular, the bus interface design requires the coprocessor to load instructions and operands, and to output results, at the correct point in the microprocessor's execution cycle. During conditional jump instructions, the coprocessor presents to the microprocessor the valid conditional test line at the correct time during the appropriate microprocessor pipeline stage. To facilitate this, the microprocessor's internal state information is available to the coprocessor on microprocessor output pins. The coprocessor uses this state information to detect bus wait states and exceptions by tracking the pipeline stage sequencing in the microprocessor. If the microprocessor causes a bus fault during a memory fetch, the coprocessor bus interface must be aware of it and may have to abort a bus operation.

A common clock is essential for synchronization between the microprocessor and coprocessor during bus operations and conditional branching on floating-point conditions. The coprocessor bus interface latches the bus input data or enables output data on the same clock edge that the processor uses. This is the most critical timing and is controlled by very simple gating to minimize potential problems. Hold times provided by the memory system and the data bus assist in providing safe margins.

To ensure that the two chips, the microprocessor and coprocessor, are able to operate together, the clock signals will be generated from a common external source and carefully routed to each chip. Within each chip, critical synchronization functions will be controlled by a minimum of logic gates. The chip IO pads and buffers will be identical on both chips to ensure compatible input and output delay times. The microprocessor's short execution cycle does not leave a very large margin for error or clock skew, so this area must be designed very carefully.

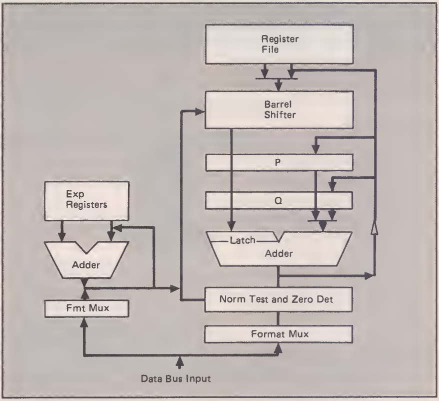


Figure 7. Coprocessor data path.

The memory interface

The high memory bandwidth required by the microprocessor led to separate instruction and operand buses. Consequently, the chip die size is determined by the input-output pad spacing, not by the number of transistors. The empty die area between the active die area and the input-output pads will be filled in a future version with on-chip instruction cache. Since all instructions are 32-bits long, the instruction memory is word addressable. But to support 8-, 16-, and 32-bit data types, the operand memory is byte addressable.

As with any high-cycle-speed microprocessor, the memory interface and the memory system are integral, and vitally important, parts of the design. Communication delays play a much more prominent role in GaAs than in silicon, compounding the problem of memory chip access delays. With an ultimate goal of a five-nanosecond execution cycle, the time it takes for a signal to travel down a wire to the memory and back becomes significant. One soon comes up against physical limits that no clever design will solve.

To retain performance, the memory system's access time must match the microprocessor's execution cycle time on the majority of memory accesses. A hierarchical memory system will be necessary for all but the smallest implementation. This approach will place a small, very fast cache near the microprocessor. For many systems, an intermediate buffer between main memory and the cache may be used to temporarily store data deleted from the cache in case it is soon needed again. Not only do physical delays affect data signals but they also exaggerate clock skew between the microprocessor and the cache. Current estimates predict that clock skew will account for 5 to 10 percent of the total cache access time.

The dual instruction and operand buses, with demultiplexed address and data lines, will greatly simplify the cache system design. Not only can separate caches be designed to suit the unique characteristics of the individual memories, but each cache can be made smaller than a single cache without decreasing the hit rate. The instruction cache will be much simpler than the operand cache due to greater locality and absence of writes. Developing an operand cache that will provide acceptable hit rates will be one of the greatest challenges of this project.

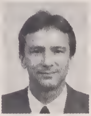
We have described the design and implementation of a 32-bit microprocessor and floating-point coprocessor in gallium arsenide using N-channel enhancement mode junction field effect transistors. We have identified areas of the architecture and design that were critical to the GaAs implementation and explained our reasons for making the decisions that we made. Both chips have been simulated using the Endot N.2 behavioral simulator (Endot, Inc.) to verify and evaluate the architecture. During 1986 a cache memory design will be evaluated with this simulator to obtain design parameters and hit rates for a cache memory system on a dual bus microprocessor. The chip level design and chip layouts will be completed by the end of this year, and processing will begin early next year. We expect to have functional chips in 1987. With improvements in the GaAs JFET currently in development, our goal is an execution cycle of 5 nanoseconds.☐

Acknowledgments

The authors appreciate the assistance they received from W. Loutensock and R. Roberts, both of the McDonnell Douglas Astronautics Company, and R. Zuleeg of the McDonnell Douglas Microelectronics Center.

References

1. R. C. Eden, A. R. Livingston, and B. M. Welch, "Integrated Circuits: the Case for Gallium Arsenide," *IEEE Spectrum,* Vol. 20, No. 12, Dec. 1983, pp. 30-37.
2. B. K. Gilbert, "Design and Performance Tradeoffs in the Use of VLSI and Gallium Arsenide in High Clockrate Signal Processors," *Proc. IEEE ICCD 84,* Port Chester, N.Y., Oct. 1984, pp. 260-266.
3. V. M. Milutinović, D. A. Fura, and W. A. Helbig, "An Introduction to GaAs Microprocessor Architecture for VLSI," *Computer,* Vol. 19, No. 3, Mar. 1986, pp. 30-42.
4. R. Zuleeg, J. K. Notthoff, and G. L. Troeger, "Double-Implanted GaAs Complementary JFETS," *IEEE Electron Devices Letters,* Vol. EDL-5, Jan. 1984, pp. 21-23.
5. R. Zuleeg et al., "Femto-Joule High-Speed Planar GaAs E-JFET Logic," *IEEE Trans. on Electron Devices,* Vol. ED-25, June 1978, pp. 628-639.
6. R. Zuleeg, "Radiation Effects in GaAs Integrated Circuits," Chapter 11 in Vol. 11, *VLSI Electronics, Microelectronic Science,* Academic Press, New York, 1985.
7. D. A. Patterson and D. L. Ditzel, "The Case for the Reduced Instruction Set Computer," *ACM Sigarch Computer Architecture News,* Vol. 8, No. 6, Oct. 1980, pp. 25-32.
8. J. L. Hennessy, "VLSI Processor Architecture," *IEEE Trans. Computers,* Vol. C-33, No. 12, Dec. 1984, pp. 1221-1246.
9. J. L. Hennessy, N. Jouppi, F. Daskett, T. R. Gross, and J. Gill, "Hardware/Software Tradeoffs for Increased Performance," *Proc. Sigarch/Sigplan Symp. Architectural Support for Programming Languages and Operation Systems,* ACM, Palo Alto, Calif., March 1982, pp. 2-11.
10. M. Katevenis, "Reduced Instruction Set Computer Architectures for VLSI," PhD dissertation, University of California, Berkeley, Oct. 1983.
11. G. Radin, "The 801 Minicomputer," *Proc. Sigarch/Sigplan Symp. Architectural Support for Programming Languages and Operation Systems,* ACM, Palo Alto, Calif., March 1982, pp. 39-47.
12. F. Chow, "A Portable, Machine-independent Global Optimizer—Design and Measurements," PhD dissertation, Stanford Univ., Stanford, Calif., 1984.
13. T. R. Gross and J. L. Hennessy, "Optimizing Delayed Branches," *Proc. Micro-15,* IEEE, Oct. 1982, pp. 114-120.
14. D. Clark and H. Levy, "Measurement and Analysis of Instruction Use in the VAX 11/780," *Proc. Ninth Annual Symp. Computer Architecture,* ACM/IEEE, Austin, Tex., April 1982.
15. R. P. Brent and H. T. Kung, "A Regular Layout for Parallel Adders," *IEEE Trans. Computers,* Vol. C-31, No. 3, March 1982, pp. 260-264.



Terrence Rasset received his BS degree in mathematics from Western Washington University in 1978 and his MS in information and computer science from the University of California, Irvine, in 1985. Since 1978 he has worked for the McDonnell Douglas Astronautics Company in Huntington Beach, California. He is currently manager of the processor advanced technology group and is responsible for the development of custom real-time processor systems and support software. His interests include advanced computer architectures for VLSI, real-time software, and application support tools. He is currently developing the architecture for an all-GaAs digital signal processor.

Roger Niederland received his BS in electrical engineering from Rutgers University in 1980. Currently employed by McDonnell Douglas Corporation in Huntington Beach, he is working in the processor development group and is group leader for integrated circuit design. Previously he was involved with the design of the MDC281, a 1750A-MIL-STD microprocessor chip set.

John H. Lane received his BS degree in physics from the New Mexico Institute of Technology in Socorro in 1955 and his MS in electrical engineering from the University of New Mexico in Albuquerque in 1961. Since 1971 he has worked for McDonnell Douglas Astronautics Company. He currently is a staff manager responsible for computer architecture and system design. Previously he was responsible for the architecture of the MDC281 MIL-STD-1750A processor chip set as well as other general-purpose and special-purpose computer architectures.

William A. Geideman received his BS in physics from Iona College, New Rochelle, New York, in 1957 and his MS in physics from Stevens Institute of Technology, Hoboken, New Jersey, in 1960. He joined McDonnell Douglas in 1973 and is currently a staff senior manager for microelectronics applications in the Astronautics Division in Huntington Beach. Prior to this he was the chief engineer for GaAs digital circuit design and fabrication in the microelectronics center. He is responsible for the application of GaAs digital circuits to internal system programs as well as for prime government contracts.

Readers may contact the authors at the McDonnell Douglas Astronautics Company, Huntington Beach, CA 92647.

CALL FOR PAPERS

1987 IEEE Symposium on Security and Privacy

April 27-29, 1987

The Claremont Hotel
Oakland, California

sponsored by
The Computer Society of the IEEE
—Technical Committee on Security and Privacy
in cooperation with
The International Association for Cryptologic Research (IACR)

The symposium will address advances in the design, implementation and application of secure computer systems. Privacy issues will also be addressed. Papers and panel session proposals are solicited in the following areas:

- Security Testing & Evaluation
- Application Security
- Network Security
- Formal Security Models
- Authentication
- Data Encryption
- Data Base Security
- Operating System Security
- Privacy Issues
- Cryptography Protocols

Papers of exceptional interest and merit will be selected for publication in a special issue of IEEE Transactions of Software Engineering in 1987.

For further information concerning the symposium contact:

Virgil D. Gligor, General Chairman
Department of Electrical Engineering
University of Maryland
College Park, Maryland 20742
(301)-454-8846 or gligor@eneevax.umd.edu. (ARPA net)

The Program Co-Chairpersons are:

David Bailey
Los Alamos National Laboratory
Mail Stop F677 OS-DO
P.O. Box 1663
Los Alamos, New Mexico 87545

Steve Lipner
Digital Equipment Corporation
305 Foster Street
MS. LTN 2-2/108
Littleton, MA 01460

The Treasurer is:

Terry Vickers-Denzel
Mitre Corporation
Burlington, MA 01730

THE COMPUTER SOCIETY OF THE IEEE

THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, INC
IEEE

Association For Computing Machinery
1987 Computer Science Conference®

February 17-19, 1987 • Clarion and Adams Mark Hotels • St. Louis, Missouri

The 1987 ACM Computer Science Conference will consist of three full days of technical sessions, featuring papers, abstracts, panels, keynote and invited presentations. The technical program will cover the spectrum of computer science interest areas, and is designed to expand the knowledge of the participants and to enhance their expertise in the fields of interest.

TECHNICAL PROGRAM

TUESDAY, FEBRUARY 17

SPECIAL SESSION ON EXPERT SYSTEMS

SESSION CHAIRMAN:
Dr. John McGregor, Murray State University
KEYNOTE SPEAKER:
Dr. Eugene Charniak, Brown University,
"EXPERT SYSTEMS AND ARTIFICIAL INTELLIGENCE"
INVITED SPEAKERS:
Dr. Donald Waterman, Rand Corporation,
"EVALUATION OF EXPERT SYSTEM BUILDING TOOLS"
Texas Instruments,
"USE OF EXPERT SYSTEMS IN THE DEFENSE INDUSTRY"
AFTERNOON –
Concurrent Sessions of Refereed Papers, Abstracts and Panels

WEDNESDAY, FEBRUARY 18

SPECIAL SESSION ON COMPUTER GRAPHICS

SESSION CO-CHAIRMEN:
Dr. Ralph Wilkerson, University of Missouri-Rolla
Doug Meyer, University of Missouri-Rolla
KEYNOTE SPEAKERS:
Dr. Edwin Catmull, Pixar
Dr. James Foley, George Washington University
INVITED SPEAKER:
Frank Dawson, McDonnell-Douglas
AFTERNOON –
Concurrent Sessions of Refereed Papers, Abstracts and Panels

THURSDAY, FEBRUARY 19

SPECIAL SESSION ON COMPUTER SCIENCE EDUCATION

SESSION CHAIRMAN:
Dr. Daniel C. St. Clair, University of Missouri-Rolla
KEYNOTE SPEAKER:
Dr. Nicholas Wirth, Swiss Federal Institute of Technology
MORNING –
Concurrent Sessions of Undergraduate Student Papers
AFTERNOON –
Concurrent Sessions of Refereed Papers, Abstracts and Panels

OTHER CSC '87 ACTIVITIES

- The Fifteenth Annual Computer Science Employment Register
- Exhibits featuring the latest products and services in computer science
- The Eleventh Annual ACM Scholastic Programming Contest

HOTEL ACCOMMODATIONS

Accommodations have been arranged for conference attendees and their guests at two luxurious downtown St. Louis properties, the Clarion and Adams Mark Hotels.

CALL FOR ABSTRACTS

Abstracts are still being accepted for CSC '87. They must be received by November 3, 1986. for information, please contact:

Dr. George Zobrist, Program Chairman
1987 ACM Computer Science Conference
Dept. of Computer Science
University of Missouri-Rolla,
Rolla, MO 65401
(314) 341-4492

SIGCSE TECHNICAL SYMPOSIUM
FEBRUARY 19-20, 1987

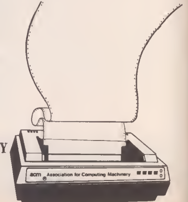
The Eighteenth ACM/SIGCSE Technical Symposium on Computer Science Education will be held February 19-20, 1987 in conjunction with the 1987 ACM Computer Science Conference. The Symposium is sponsored by the ACM Special Interest Group on Computer Science Education (SIGCSE). Refereed papers and panel sessions on timely issues of interest to computer science and information processing educators will be highlighted.

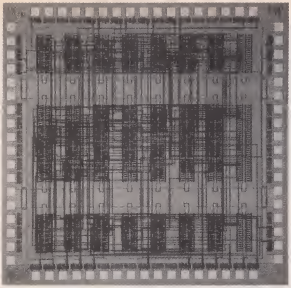
FOR DETAILS CONTACT:

CSC '87 Co-Chairmen
Arlan R. DeKock
John Hamblen
Dept. of Computer Science
University of Missouri-Rolla,
Rolla, MO 65401
(314) 341-4491

SIGCSE Chairman
Dan C. St. Clair
Computer Science Department UMR
UMSL Campus
8001 Natural Bridge Road
St. Louis, MO 63121-4499
(314) 553-5440

ASSOCIATION FOR COMPUTING MACHINERY
11 WEST 42 STREET,
NEW YORK, NY 10036
(212) 869-7440


acm Association for Computing Machinery



An 820-gate, programmable function interval timer (PROFIT) circuit implemented on a TI 1K GaAs gate array. (Photo courtesy of Texas Instruments.)

Reduced Instruction Set Architecture for a GaAs Microprocessor System

Eric R. Fox, Kenneth J. Kiefer, Robert F. Vangen, and Shaun P. Whalen

Control Data Corporation

RISC architecture characteristics of low hardware requirements, large register files, and pipelining accommodate the gate count constraints and preserve the inherent speed of GaAs technology.

Gate densities that permit the integration of an entire microprocessor on a single chip have been reached in GaAs technology. A reduced instruction set computer, or RISC, architecture is well suited to implementation in GaAs because of its low hardware requirements. The large register file and pipelined architecture typical of RISCs$^{1-2}$ complement the high on-chip gate speeds of GaAs by reducing off-chip communication. In late 1984 Texas Instruments and we at Control Data, under DARPA sponsorship, began a one-year project to develop a GaAs microprocessor system with a RISC architecture. The system we developed consists of a CPU, a floating-point coprocessor, or FCOP, a memory management unit, or MMU, and a cache. The streamlined architecture minimizes latencies between instructions while allowing for parallel operation between the CPU and the FCOP. The MMU manages the cache to provide a high hit rate. The tight coupling between CPU, FCOP, and MMU achieves a peak throughput of 200 MIPS.

Technology

In order to make it possible to successfully build a fully functional chip, we had established that a maximum gate count of 10,000 was necessary. Also in order to obtain the fastest possible machine cycle time interaction between GaAs chips had to be kept to a minimum to preserve the inherent speed of the GaAs gates. This required that we put entire logical functions on one chip. To place the CPU on one chip we had to eliminate control gates as much as possible while implementing as many on-chip registers as possible to minimize off-chip activity. The RISC architecture was a natural choice to satisfy these requirements.

When ready for large-scale production the CPU, COP, and MMU circuits are to be fabricated using GaAs heterojunction integrated injection logic, or HI^{2}L, technology. Analysis shows this is a good choice for the implementation of the GaAs microprocessor system. The HI^{2}L nand gates require only one transistor, which permits high gate density and smaller chip size. With a substrate ground and only one voltage bus needed, power distribution is greatly simplified. With HI^{2}L good control of logic thresholds over temperature and voltage ranges is obtainable since doping levels have little effect on thresholds. Control over gate speeds may be individually programmed by selecting the value of the base resistor with a single mask step. This flexibility permits designing very fast gates as needed in critical paths while using slower and lower power gates where they can be tolerated. This is important to allow reaching the 10,000 gate limit without using excessive power.

0018-9162/86/1000-0071$01.00 © 1986 IEEE

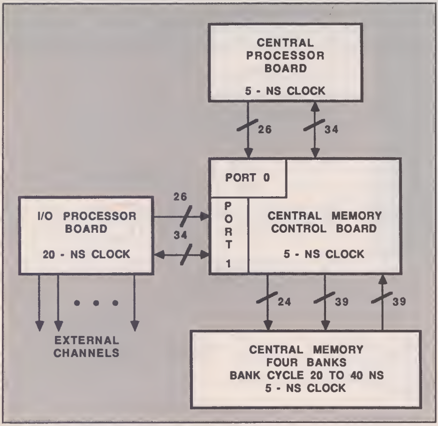


Figure 1. GaAs microprocessor system logical configuration.

System overview

Figure 1 is a logic diagram of the GaAs microprocessor system. The system has full 32-bit data paths throughout and supports a 64M-byte virtual and real address space. The objective of the system is to operate using a 5-ns clock cycle, thereby attaining a 200-MIPS peak execution rate. All elements in the system operate from the same clock system. The CPU chip is the heart of the system and generates all memory addresses for both instructions and operands. The CPU, FCOP, and MMU chips each are capable of executing their own specific instructions using a common instruction bus. Each chip controls a six-stage pipeline so an orderly exit and return can be taken to an interrupt routine. The MMU supports a fully segmented and paged virtual memory system. It also controls independent instruction and data caches, which allow parallel access. The I/O processor and CPU have separate ports on the central memory control, or CMC, board to a multibank central memory to minimize contention and conflict delays.

The GaAs microprocessor system central processor board consists of a CPU chip, a FCOP chip, two MMU chips, two processor board interface chips, along with several RAMs and support chips utilized as cache memories. The CPU, FCOP, and MMU chips are GaAs custom circuits using approximately 10,000 gates and 136 I/O pins. The processor board interface chips are GaAs gate arrays. The cache memories are built using 256- × 4-bit and 1K- × 4-bit GaAs RAMs.

Error detection logic is not included in the chips. Parity protection is included on both the cache tag and cache data memories. The central memory is protected by a single-bit error correction double bit error detection, or SECDED, code that is generated and checked by the CMC board. When a parity error is detected, an error interrupt is sent to the CPU, FCOP, and MMU chips.

Instruction set architecture, or ISA

An initial instruction set was developed based on simulation statistics in the literature,[1,3] and from our own studies. The GaAs microprocessor system instruction set design was modified by the introduction of a core instruction set requirement. A core instruction set was defined at an intermediate level between the hardware-dependent GaAs microprocessor system instruction set and a high-level language. Once a program is compiled to the core instruction set, a hardware-dependent translator transforms the code to the machine instruction set and performs any machine-dependent optimizations. The focus of the GaAs microprocessor system instruction set was changed from emulating macro functions in high-level language statements to supporting macro functions in the instructions of the core instruction set.

The instruction set is divided into three parts corresponding to the three elements of the GaAs microprocessor system: the CPU, the FCOP and the MMU (Figures 2a, 2b, and 2c).

Floating-point operations cannot be executed in a single cycle even with fast GaAs logic. Floating-point operations may be partitioned for single machine cycle execution. The FCOP instruction set consists of instructions that perform parts of a given floating-point operation. For example, single-precision floating-point add is performed by executing the following:

DADDS	Denormalize operand with smaller exponent. Add the mantissas of the two operands. Compare exponents of the two operands to give result exponent. Convert mantissa sum/difference from two's complement to sign magnitude format.
NRMRNS	Normalize and round result mantissa to single precision.

The three instruction sets provide an integrated system that includes floating-point and memory management capability. The CPU performs instruction and operand addressing for the system. Instructions fetched by the CPU may be executed by the CPU, FCOP, or the MMU. The CPU controls the system interrupt behavior. Upon receiving an interrupt, the FCOP and MMU suspend their pipelines while the CPU performs a context switch. The MMU and FCOP are informed of changes in the address privilege, address mapping, and cache usage by the monitor, map, and cache bypass signals. The CPU, FCOP, and MMU may communicate by passing data directly from one to the other. Conditions evaluated on the FCOP are available to the CPU via the coprocessor condition signal.

The architecture allows for a maximum of four coprocessors in the system. Other coprocessor instruction sets must support memory access and data movement between processors in the same way that these operations are supported in the CPU, FCOP, and MMU.

Single-precision integer add	ADD,SRC1,SOURCE2*,DEST	Load word indexed	LDW,SOURCE2[SRC1],DEST
Single-precision integer add with carry	ADDC,SRC1,SOURCE2,DEST	Load halfword indexed	LDH,SOURCE2[SRC1],DEST
Single-precision integer subtract	SUB,SRC1,SOURCE2,DEST	Load byte indexed	LDB,SOURCE2[SRC1],DEST
Single-precision integer subtract with carry	SUBC,SRC1,SOURCE2,DEST	Store word indexed	STW,SRC2,SYMBOL[SRC1]
Single-precision integer subtract reverse	SUBR,SRC1,SOURCE2,DEST	Store halfword indexed	STH,SRC2,SYMBOL[SRC1]
Single-precision integer multiply step	MSTEP,SRC1,SRC2,DEST	Store byte indexed	STB,SRC2,SYMBOL[SRC1]
Logical AND	AND,SRC1,SOURCE2,DEST	Store word direct	STW,SRC2,LITERAL
Logical inclusive OR	OR,SRC1,SOURCE2,DEST	Store halfword direct	STH,SRC2,LITERAL
Logical exclusive OR	XOR,SRC1,SOURCE2,DEST	Store byte direct	STB,SRC2,LITERAL
Shift right arithmetic by one	SRA,SOURCE2,DEST	Trap conditional	TRAPCND,COND,TRAP_CODE
Shift right logical by one	SRL,SOURCE2,DEST	Jump indexed	JMP,SOURCE2[SRC1]
Move register or immediate	MOV,SOURCE2,DEST	Call indexed	CAL,SOURCE2[SRC1]
Implicit source move	MOV,IMPL_SRC,DEST	Return from trap	RETT,LITERAL,DST
Implicit destination move	MOV,SOURCE2,IMPL_DEST	Branch conditional	BRC,COND,SOURCE2
Load high immediate	LDHI,HO_BITS,DEST		

(a) *SOURCE2 is either a sign extended literal or the register indicated by src2.

Purge page map	PMAP
Purge cache	PCACH
Load inst mmu page map	LDW,SOURCE2[SRC1],MDEST
Load oper mmu page map	LDW,SOURCE2[SRC1],MDEST
Implicit Source move reg from mmu to cpu	MOV,MIMPL_SRC,DEST
(b) Implicit Destination move reg from cpu to mmu	MOV,SRC2,MIMPL_DEST

No operation	NOP
Single-prec denormalize, add, convert to sign magnitude	DADDS,CSRC1,CSRC2
Single-prec denormalize, sub, convert to sign magnitude	DSUBS,CSRC1,CSRC2
Double-prec denormalize, add	DADDD,CSRC1,CSRC2
Double-prec denormalize, sub	DSUBD,CSRC1,CSRC2
Convert to sign magnitude, lower half of double prec	CSML
Convert to sign magnitude, upper half of double prec	CSMH
Single-prec convert integer to floating point	FLDATS,CSRC1
Double-prec convert integer to floating point	FLDATD,CSRC1
Single-prec normalize and round to nearest	NRMRNS
Double-prec normalize and round lower half to nearest	NRMRNDL
Round double-precision upper half	RNDH
Single-prec multiply floating start	MFSTS,CSRC1,CSRC2
Single-prec multiply integer start	MISTS,CSRC1,CSRC2
Single-prec multiply step	MSTEP
Single-prec multiply floating finish	MFINFS
Single-prec divide floating start	DFSTS,CSRC1,CSRC2
Single-prec divide integer start	DISTS,CSRC1,CSRC2
Single-precision divide step	DSTEP
Single-precision divide floating finish	DFINFS
Implicit Source move reg from fcop to cpu	MOV,CIMPL_SRC,DEST
Implicit Destination move reg from cpu to fcop	MOV,SRC2,CIMPL_DEST
Load register file high	LDHI,SOURCE2[SRC1],CDEST
Load register file low	LDLO,SOURCE2[SRC1],CDEST
Load register file, convert to single prec	LDCV,SOURCE2[SRC1],CDEST
Store register file high	STHI,CSRC1,SYMBOL[SRC1]
Store register file low	STLO,CSRC1,SYMBOL[SRC1]
Store register file, convert from single prec	STCV,CSRC1,SYMBOL[SRC1]
P1 register high to register H	MOV,P1,H
P1 register low to register L	MOV,P1,L
(c) Source 1 to P1, Source 2 to P2	MOV,CSRC1/CSRC2,P1/P2

Figure 2. (a) CPU instruction set; (b) MMU instruction set; (c) FCOP instruction set.

Pipeline

Pipelining is an effective method for reducing interchip communication while increasing performance. The GaAs microprocessor system was initially designed with a four-stage pipeline. Fetching one instruction every cycle gives a high instruction bandwidth while permitting the instruction format to be simply encoded.

The cache memory access could not support the 5-ns memory cycle required by the four-stage pipeline. A pipelined memory preserved the 5-ns cycle time by permitting the memory to be accessed over two cycles. The resulting six-stage pipe consists of instruction fetch cycle 1 (I1), instruction fetch cycle 2 (I2), ALU execute (EX), memory access cycle 1 (M1), memory access cycle 2 (M2), and write register file (WR). (See Figure 3.)

The nonmemory access pipestages EX and WR execute the instruction and write the register file. The memory access pipestages I1, I2, M1, and M2 put the memory address out during the I1 or M1 cycle and latch the returning instruction or operand data during I2 and M2. The memory chips have pipeline registers integrated on-chip with the memory array to support pipelined access. The pipelined memory access increased the complexity of the design but decreased the cycle time by 39 percent.

The GaAs microprocessor system has some hardware pipeline interlocks. Data dependencies between register-to-register instructions are resolved in hardware. (See Figure 4.)

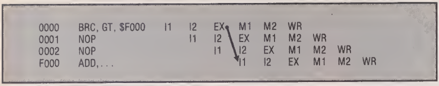
Scheduling of operations in the instruction slots after a branch is not provided by the hardware. On a branch, the software must ensure that instructions in the pipe after the branch do not depend upon the destination of the branch. If no benign instruction can be found to fit into the delay

I1	I2	EX	M1	M2	WR					
	I1	I2	EX	M1	M2	WR				
		I1	I2	EX	M1	M2	WR			
			I1	I2	EX	M1	M2	WR		
				I1	I2	EX	M1	M2	WR	
					I1	I2	EX	M1	M2	WR

Figure 3. Final GaAs microprocessor pipeline.

ADD, RD, R1, R2	R2 = R1 + RD	I1	I2	EX	M1	M2	WR	
SUB, R2, R3, R4	R4 = R2 − R3		I1	I2	EX	M1	M2	WR

Figure 4. Data dependencies. Although the ADD does not write **R2** until the WR pipestage, the hardware shortstopping logic provides the sum of R1 and R0 to the SUB instruction in its EX pipestage.



```
0000   BRC, GT, $F000   I1   I2   EX   M1   M2   WR
0001   NOP                   I1   I2   EX   M1   M2   WR
0002   NOP                        I1   I2   EX   M1   M2   WR
F000   ADD,...                         I1   I2   EX   M1   M2   WR
```

Figure 5. Branch delays. The branch delay for a six-stage pipeline is two. The two instructions after a branch are always executed since these instructions are already in the pipeline when the branch is evaluated during EX.

slot, then it must be filled with a no operation instruction (nop). (See Figure 5.)

The pipeline organization has a direct impact on performance by determining the number of delay slots after a branch. The number of branch delays in the four-stage pipeline is one. The number of branch delays for the six-stage pipeline is two. Although it is more difficult to fill two branch delays than one with useful instructions, simulations show that the six-stage pipeline net MIP rate is 39 percent better than the four-stage pipeline.

The interrupt convention followed in the GaAs microprocessor system architecture stops interrupted instructions from completing, shuts down the pipeline, fixes the cause of the interrupt, and restarts the interrupted instruction stream. Shutting down the pipeline involves two tasks: saving the addresses of the interrupted instruction stream and saving the state of the system. The addresses of the instructions in the M2, M1, and EX pipestages are saved to allow the instruction stream to be restarted upon return from the interrupt. The state of the system is preserved by preventing the interrupted instructions from altering permanent registers, the register files, or memory.

Returning from an interrupt is performed by jumping to the addresses of the three instructions saved when the interrupt occurred. The return from interrupt routine also restores the system to its original status.

Software

All software is written in a high-level language such as Pascal or Ada. This software then passes through a compiler, translator, reorganizer, assembler, and linker/loader before execution, as shown in Figure 6. The process begins with the compiler, which translates programs written in Pascal or Ada into the MIPS Core ISA assembly language.

The translator searches the core ISA code and by using a combination of macros, procedures, and machine code sequences translates the core ISA to machine ISA. For most instructions, the mapping from the core instruction to low-level instruction is a direct translation into the machine instruction.

The reorganizer takes the translated machine ISA and repositions instructions. The reorganization of instructions results in more efficient program code. Code reorganization packs the branch delay slots with useful instructions when possible. Also, memory reference instructions cannot be immediately followed by instructions that use the memory reference; they must be delayed two instructions. In such cases, the translator will

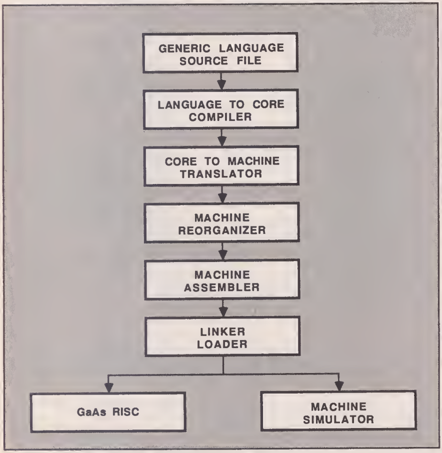

GENERIC LANGUAGE SOURCE FILE
LANGUAGE TO CORE COMPILER
CORE TO MACHINE TRANSLATOR
MACHINE REORGANIZER
MACHINE ASSEMBLER
LINKER LOADER
GaAs RISC
MACHINE SIMULATOR

Figure 6. Support software overview.

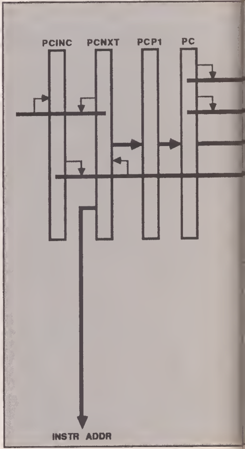


Figure 7. CPU data path.

load the pipeline with nops, relying on the reorganizer to eliminate as many as possible by replacing them with position-independent instructions.

The assembler and linker and loader complete the translation process, either producing a loadable simulator file or a binary file executable by the GaAs microprocessor system.

Central processor unit, or CPU

The CPU incorporates interfaces, data paths, and control to execute efficiently the CPU instruction set. The CPU has dedicated buses for operand and instruction memory access. The memory interface consists of a 24-bit wide instruction address bus, a 32-bit wide instruction data bus, a 26-bit wide operand address bus, and a 32-bit wide operand data bus. Instructions are word addressed. Operands are byte addressed. Operand data may be in byte, halfword, or word format. The control interface provides status, memory read/write control, and interrupt information to the system or the CPU.

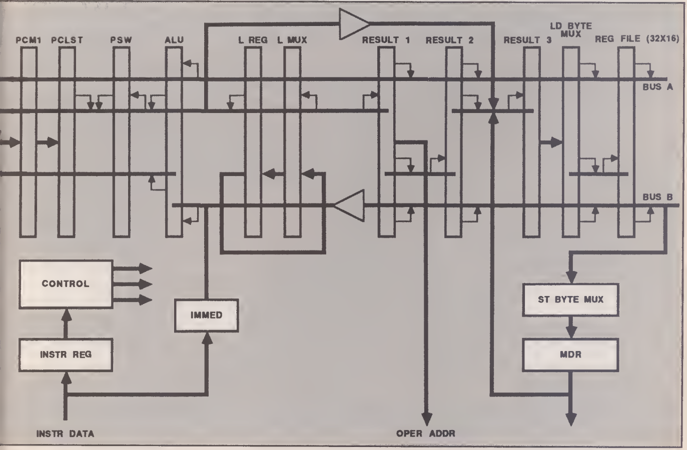
The CPU execution model (read two operands, perform an ALU operation, write the result) is supported by a 32-bit wide data path. The major data path elements are the register file, the temporary registers, the ALU, the PSW, and the program counters. (See Figure 7.)

Data flows through the data path from the register file through the ALU and back to the register file. An instruction is fetched using the address in PCNXT during I1 and I2. During EX, the register file is read onto the a and b buses. Alternatively, the contents of result1, result2, or result3 may be "shortstopped" to one or both buses. The ALU performs the arithmetic, logical, or shift operation. The ALU result is latched by the result1 register on ALU and memory instructions, sent to the PCNXT on control transfer instructions, or latched by the PSW on an implicit move to the PSW instruction. During M1, the result1 contents are sent to the result2 register. The result1 contents may be used as an address for an operand cache access. The condition codes, if set by the instruction, are evaluated during M1 and latched in the PSW. During M2, the result3 register latches the ALU output, the result2 contents, or the memory interface.

During WR, the register file is written by the result3 register. During each pipestage, the instruction address is passed from one program counter to the next. Therefore, PCNXT, PCP1, PC, PCM1, and PCLST hold the addresses of the instructions in the I1, I2, EX, M1, and M2 pipestages, respectively.

The streamlined instruction set and pipeline structure result in a simple state machine. The state machine has six states. The initial state after an interrupt is the TR1 state. This state enables only the I1 pipestage. All interrupts except reset are disabled upon entering TR1 state. The successive states enable the next pipestages in the pipeline until the state machine returns to the NORM state, where all six pipestages are executing. The simple state transitions permit a small state machine. (See Figure 8.) Memory waits suspend the state machine until the memory is ready.

Context switches for interrupts complicate a load/store architecture. The register file is the destination or source of loads and stores. The program counters must be accessed directly on an interrupt to avoid corrupting the register file. To avoid the increased complexity and gate count of store and load program counter instructions, the state machine and the primitive instructions synthesize these functions. On an interrupt, the program



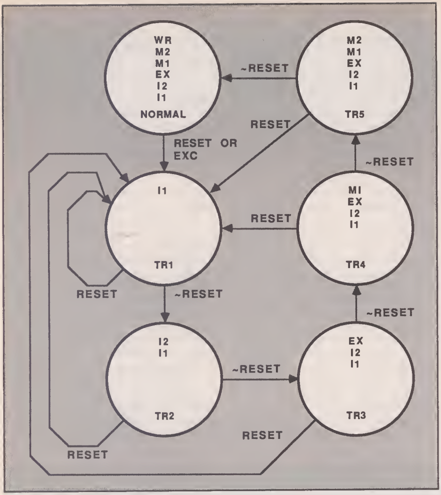


Figure 8. State machine. State transitions are determined by the reset, interrupts, and interrupt enables. EXC is the sum of product terms involving the interrupts and their respective enables.

	I1	I2	EX	M1	M2	WR											
		I1	I2	EX	M1	M2											
			I1	I2	EX	M1											
				I1	I2	EX											
					I1	I2											
						I1											
STWD,R0,$0000							I1	I2	EX	M1	M2	WR					
STWD,R1,$0004								I1	I2	EX	M1	M2	WR				
STWD,R2,$0008									I1	I2	EX	M1	M2	WR			
MOV,PCLST,R0										I1	I2	EX	M1	M2	WR		
MOV,PCLST,R1											I1	I2	EX	M1	M2	WR	
MOV,PCLST,R2												I1	I2	EX	M1	M2	WR
	t	t+1	t+2	t+3	t+4	t+5	t+6	t+7	t+8	t+9							

Figure 9. Pipestage activity during an interrupt. An interrupt occurs during the pipestage at t+5. All pipestages are disabled except for I1, which begins fetching the interrupt service routine at location zero. The first six instructions of the interrupt routine move registers R0, R1, and R2 to memory and move the program counters into the register file.

counters are inhibited from changing until three registers in the register file have been saved. After three locations have been freed in the register file, the program counters may be moved there and then to memory. (See Figure 9.)

A return from interrupt requires that the return instruction addresses be fetched from memory without going through the register file. The return from trap instruction loads the return address from memory into a temporary register, where it may be used by a jump instruction to transfer control. The temporary register is not written to the register file. (See Figure 10.)

The return from trap returns the system to its original context without corrupting the register file.

FCOP

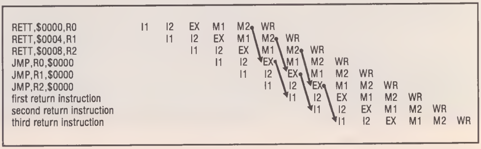
The FCOP, or floating-point coprocessor, contains the hardware to execute the floating-point portion of the instruction set. The FCOP also contains hardware support for fast integer multiply and divide. The FCOP conforms to IEEE standard floating-point format for single-and double-precision operands.

The FCOP interface reflects its role as a coprocessor in the system and receives instructions and operands over two dedicated 32-bit buses. The FCOP control interface contains the status inputs, interrupt signals, memory wait conditions, the coprocessor condition, and two signals to indicate the binary code of a particular FCOP if more than one coprocessor is in the system.

The data path of the FCOP consists of the following major elements: 4- × 64-bit register file, exponent control, 64-bit shift unit, 34-bit arithmetic unit, and FCOP PSW. (See Figure 11.)

Floating-point operations require a more complex execution model than integer operations. The FCOP hardware is used iteratively to perform the various parts of a floating-point operation. For example, a double-precision floating-point addition begins by reading the two

Figure 10. Pipestage activity on a return from trap. The RETT instruction fetches the first return address and intends to write it to register RO. The JMP instruction notes that the destination of the RETT is the same as its source and shortstops the contents of the temporary register to be used as the target address of the jump.



RETT,$0000,R0	I1	I2	EX	M1	M2	WR								
RETT,$0004,R1		I1	I2	EX	M1	M2	WR							
RETT,$0008,R2			I1	I2	EX	M1	M2	WR						
JMP,R0,$0000				I1	I2	EX	M1	M2	WR					
JMP,R1,$0000					I1	I2	EX	M1	M2	WR				
JMP,R2,$0000						I1	I2	EX	M1	M2	WR			
first return instruction							I1	I2	EX	M1	M2	WR		
second return instruction								I1	I2	EX	M1	M2	WR	
third return instruction									I1	I2	EX	M1	M2	WR

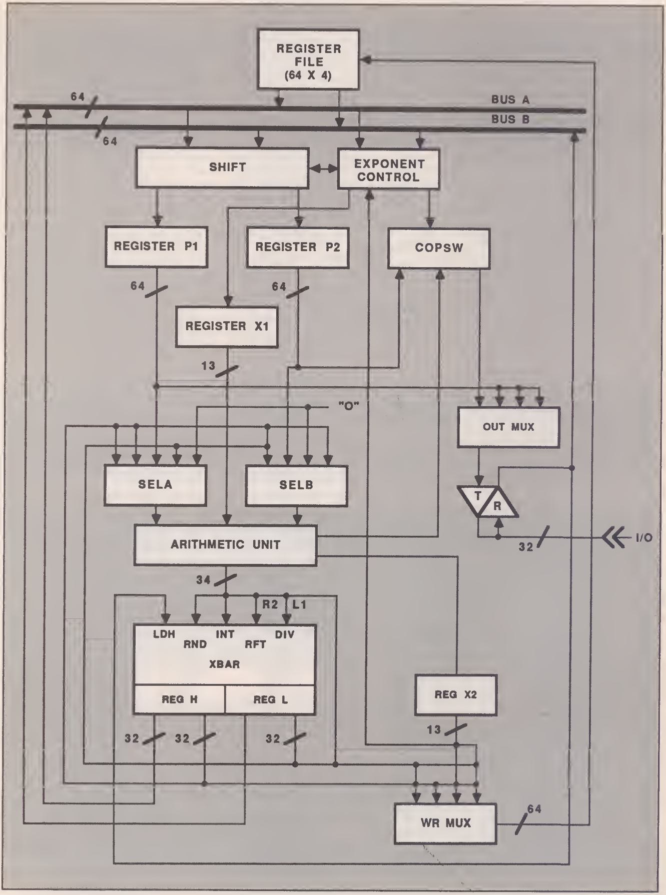


Figure 11. FCOP data path.

operands out of the register file during the EX pipestage. The exponents of the operands are compared and the appropriate shift amount sent to the shifter for denormalization. The shifter denormalizes the mantissa with the smaller exponent. The preliminary result exponent is latched in the X1 register and the mantissas in the P1 and P2 registers. During the M1 pipestage, the arithmetic unit adds the low parts of the two mantissas. The carry out is saved and used during M2 to add the upper parts of the two mantissas. The next instruction converts the lower part of the arithmetic unit two's complement output to sign magnitude. The next instruction uses the arithmetic unit in the M2 pipestage to convert the upper part of the result mantissa to sign magnitude form. The sign magnitude mantissa is normalized in the next instruction during the M1 pipestage. During M2, the low part of the mantissa is rounded. The final instruction uses the arithmetic unit during the M2 pipestage to round the upper part of the normalized mantissa, reformat the result into double-precision format with the exponent, and write it to the register file. The floating-point condition codes in the PSW are updated in different pipestages depending on whether they are the result of a shift or add operation.

The FCOP typically cannot issue an instruction every cycle because of resource conflicts between instructions. These instruction cycles must be filled with nops if no other useful instructions can be scheduled. These nop slots provide the opportunity for the execution of CPU or MMU instructions in parallel with the FCOP's operation. The translator/reorganizer is responsible for scheduling CPU and FCOP instructions for maximum parallelism.

Cache memory

The GaAs microprocessor system CPU will require a 32-bit instruction word every 5 ns. Program statistics have shown that up to one-third of all instructions executed are load or store instructions. In order to avoid contention at the cache memory, a two-port cache would be required. But even if this were done, instructions and operands could contend for the same cache locations, reducing the cache effectiveness. For these reasons, separate instruction and operand caches are defined to allow parallel accessing of instructions and operands without contention.

When a cache miss is encountered, an eight-word block will be fetched by the MMU and processor interface. During the cache fetch process, the CPU will be in a wait state and will not proceed until the cache fetch is complete. When a store instruction is executed, the data will always be written to central memory. If the store instruction encounters a hit in the cache, the word will also be stored in the appropriate cache cell. Use of the instruction and performance simulator showed this implementation would improve the hit rate of the operand cache by 45 percent. Since the cache does not monitor I/O writes to central memory, the operating system software ensures that stale cache data is not created by I/O writes. To assist in this operation, the cache can be cleared with a purge cache instruction, and bypassed at the direction of the CPU. The cache memory addressing mechanism is shown in Figure 12. The cache size of 1K words and single element organization were chosen as a trade-off between optimum system clock cycle time, expected available circuits, and satisfactory hit ratios.

The cache contains 128 single element sets of eight words each. In the direct mapped technique, bits 5-11 of the virtual address are used to index the tag memory sets that contain the top 14 bits of a virtual address. The tag entry from the selected set is compared with the high-order 14 bits of the virtual address to determine a hit or miss condition. The same address bits along with bits 2-4 are used to read the proper word from the data memory. If a hit condition occurs, the data word read is passed on the data bus to its destination. If a miss occurs, the GaAs microprocessor system is put into a wait state while the real memory address is used to fetch the missing word into the cache. The instruction and operand caches are separate but are identical in design.

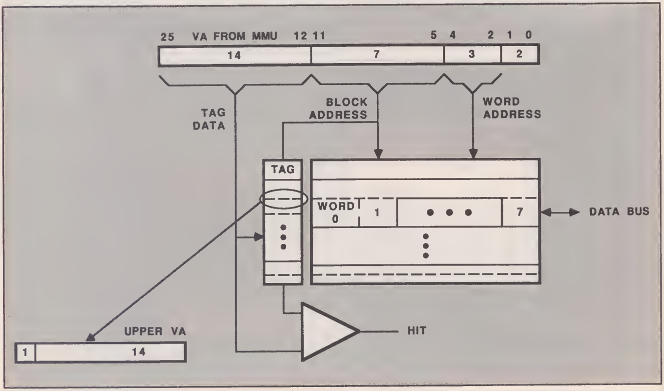


Figure 12. Cache memory operation.

Each cache is controlled by a separate MMU and each has its own address and data path so parallel referencing of instructions and data can be accomplished. The cache memories are designed to be accessed in two pipelined clock cycles. Each cache contains 1K 34-bit entries organized as 128 one element sets with eight words per set.

Process to virtual address mapping

The virtual address mapping is taken from that utilized in the MIPS processor built at Stanford.[4] When the system is in mapped mode, the MMU translates the process address received from the CPU to a virtual address using the process identifier, or PID, and mask, or MA, register contents. The translation logic substitutes the PID register contents into the high-order bits of the process address for as many bits as are set in the MA register. Since the PID register is limited to 12 bits, the smallest segment permitted by a process is then 4096 words. In the process address translation to a virtual address, the top n bits of the address are removed and replaced by an n bit PID. During the field length check the top $(n + 1)$ bits of the process address are checked to be all zeros or all ones. The value n is given by the number of bits set in the MA register. Since the process address is 26 bits, the accessible portion of the process address space is split into the low 2^{25-n} bytes and the high 2^{25-n} bytes. An attempt to access bytes outside of this range results in a field length error. The field length error causes an MMU interrupt and a field length error bit is set in the MMU PSW register along with the virtual address reference attempted.

Page map operation

The page map provides for virtual address to physical memory address translation. This translation is done through a set associative page map that contains entries for the most recently used pages. The MMU will interrupt the CPU when a page map miss is encountered. The CPU will search central memory and load the page map with the appropriate entry. During this search, the CPU may encounter a page fault that will require the page to be loaded from virtual address space to a physical memory address. The page map may be preloaded using CPU instructions.

The page map address mapping mechanism is shown in Figure 13. The page size is 2K bytes and is fixed. The map contains 8 two-element sets that provide the real memory address. In the set associative technique, the low order three bits of the page number are used to index a pair of tag elements. The tag entry from each element is compared with the high-order 15 bits of the virtual address. The two comparisons are made in parallel.

If a match is found, the real memory address is formed from the page offset and the 15 bit address from the matching real memory address element. If a match on the tag memory contents is not made, a

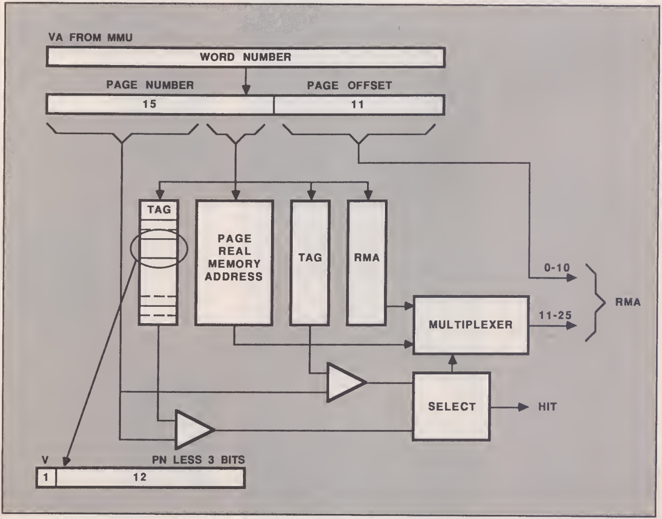


Figure 13. Page map operation.

Table 1. GaAs microprocessor simulation results.

Benchmark	Gibson mix number	Sieve prime search	String	Linked list
e = Executed instructions	716	7474	892	277
n = Number of nops	215	2290	256	114
L = Number of loads	65	258	66	32
i = Instruction cache hit rate	93%	99%	99%	96%
o = Op-cache hit rate	91%	96%	95%	94%
R = Rate (1) with simulated cache hit rate average = 91.0 MIPS	65.2 MIPS	117.4	117.0	67.2
R = Rate (2) with 100% cache hit rate average = 134.7 MIPS	139.9 MIPS	138.7	142.6	117.7

$$(1)\quad R = \frac{e-n}{[e+(1-i)\,ce+(1-o)\,cL]\,t} \qquad (2)\quad R = \frac{e-n}{et}$$

c = Cache service delay = 16 cycles
t = cycle time = 5 ns

page map miss fault results that results in a MMU interrupt signal being sent to the CPU, COP, and paired MMU. The MMU interrupt signal is used by the MMU to trap the page number of the virtual address, or VA, that was missed or had an access violation in the MMU PSW register. The page map can be bypassed at the direction of the CPU and can be purged using the purge page map instructions.

Performance summary

The GaAs microprocessor system can operate with a 200-MHz clock rate because of the efficiency of its highly pipelined architecture, but the average system net throughput, as shown in Table 1, is less than 200 MIPS because of the CPU pipelining that requires nops and because of limited operand and instruction MMU cache hit rates. During each machine cycle, stages of six different instructions are executing, thus allowing up to one instruction completion per clock. One instruction must be fetched from memory every clock cycle, with simultaneous access to required operands via the operand data bus for maximum net throughput. To minimize this memory bottleneck, the two GaAs MMUs are each used to manage a cache. The limits on memory bandwidth show up in the instruction and operand cache hit rates, and the MMU time required to service a cache miss. The average GaAs microprocessor system net throughput then is 200 MIPS, reduced by 32 percent because of required nops, and then reduced by another 32 percent because of the memory bandwidth limits. This results in an average net throughput of 91 MIPS, as shown in the benchmark simulation results of Table 1. Design enhancements are being made to the system to improve the average net throughput by reducing the number of nops in the reorganized code and improving the memory bandwidth.

In summary, the RISC system architecture exploits the high speed aspects of GaAs technology to achieve up to one instruction completion every five ns, while demanding a minimal number of gates per chip. The increased demands on memory bandwidth caused by the RISC approach are answered by including two GaAs caches. Through a balanced system design philosophy, a complete system architecture was fitted to the constraints of GaAs technology. ☐

Acknowledgments

Funding for the GaAs Microcomputer was provided by the Defense Advanced Research Projects Agency, or DARPA, under contract 84-F-167880; Texas Instruments; and Control Data Corporation.

References

1. D. Patterson and C. Sequin. "RISC I: A Reduced Instruction Set VLSI Computer," *Proc. 8th Ann. Symp. Computer Architecture,* May 1981, pp. 443-457.
2. J. Hennessy et al. "The MIPS Machine," *Proc. Compcon,* Feb. 1982, p. 107.
3. J. Hennessy et al., "Hardware/Software Tradeoffs for Increased Performance," *Proc. Symp. Architectural Support for Programming Languages and Operating Systems,* March 1-3, 1983, pp. 2-12.
4. S. A. Przybylski et al., *Organization and VLSI Implementation of MIPS,* Stanford Computer Systems Laboratory tech report no. 84-259, April 1984, pp. 8-10.

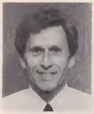
Eric R. Fox is currently working on the GaAs microprocessor system demonstration and test hardware and software. Prior experience includes design of a VLSI chip, a CPU context cache, a CPU arithmetic unit, and a portion of a floating-point firmware package.

Fox is a member of Tau Beta Pi and Eta Kappa Nu, has received a Control Data Employee Excellence Award, and holds a BSEE from Purdue University.

Kenneth J. Kiefer is a technical consultant at Control Data Corporation. He is currently involved in developing architectures for Avionics

Computers and in applying GaAs technologies to these architectures. Prior to taking this position, he worked in the development of high-performance mainframes at Control Data.

Kiefer holds a BA from Loras College, Dubuque, Iowa, and a BSEE from the University of Iowa, Iowa City.



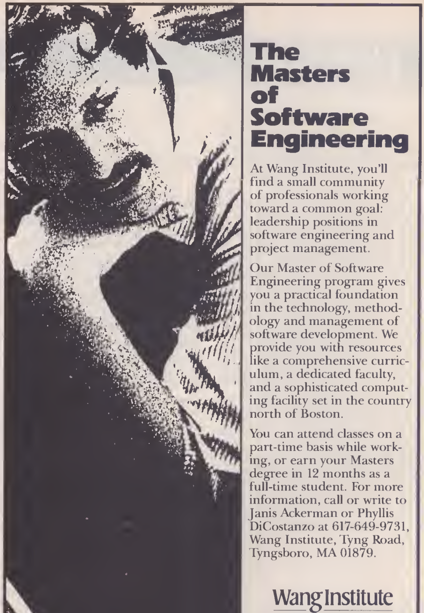
Robert F. Vangen is the GaAs Microprocessor Project Manager, and is also on the design team to develop a vector coprocessor for the GaAs microprocessor system. Present interests include high-performance computer architecture and signal processing. Prior professional experience includes development of a CAD system for design of integrated circuits. He has also designed computer peripherals, digital communication systems, and airborne computers.

Vangen has been granted one patent, is a member of Eta Kappa Nu, and has a BSEE with distinction and an MBA, both from the University of Minnesota.

Shaun P. Whalen is currently an electrical engineer with the Avionics Computer Architecture department of Control Data Corporation. His technical interests include streamlined computer architectures for high-level languages.

Whalen is a member of IEEE, Eta Kappa Nu, and Tau Beta Pi. He received his BS degree from the University of Notre Dame in 1983 and his MS degree from the University of California, Berkeley, in 1984.

Readers may write to the authors at Control Data Corporation, 8800 Queen Ave., South, Minneapolis, MN 55440.



The Masters of Software Engineering

At Wang Institute, you'll find a small community of professionals working toward a common goal: leadership positions in software engineering and project management.

Our Master of Software Engineering program gives you a practical foundation in the technology, methodology and management of software development. We provide you with resources like a comprehensive curriculum, a dedicated faculty, and a sophisticated computing facility set in the country north of Boston.

You can attend classes on a part-time basis while working, or earn your Masters degree in 12 months as a full-time student. For more information, call or write to Janis Ackerman or Phyllis DiCostanzo at 617-649-9731, Wang Institute, Tyng Road, Tyngsboro, MA 01879.

Wang Institute
OF GRADUATE STUDIES

Name

Business Name

Business Address

Business Phone Home Phone

Your current status:
☐ Software Professional
☐ Student ☐ Other

Years of Software Development Experience: ____

Wang Institute is accredited by the New England Association of Schools and Colleges and is an Equal Opportunity institution.

COM 10/86

Reader Service Number 2

Researchers fabricate composite silicon/GaAs semiconductor material

Straddling the debate on the place of traditional silicon versus high-speed gallium arsenide in semiconductor products of the future, researchers at the University of Illinois have discovered a process that allows them to successfully fabricate a composite material from the two substances. The result, the discoverers hope, will be integrated circuits with the speed and light-generating properties of gallium arsenide but the durability and economy of silicon.

A team led by Illinois professor Hadis Morkoc uses the technique of molecular-beam epitaxy to deposit thin films of gallium arsenide and other semiconductor materials onto silicon bases. Gallium arsenide substrates are currently formed from successive layers of gallium, arsenic, and other materials on a gallium arsenide base. Although extremely fast, radiation resistant, and suitable for fabricating optoelectronic devices, gallium arsenide suffers from relatively high cost, fragility, and poor heat dissipation. Furthermore, gallium arsenide substrates are susceptible to dislocations in the lattice geometry or arrangement of atoms in the material.

Morkoc and his colleagues solved this problem by depositing the gallium arsenide onto the silicon base in an arrangement called a "strained-layer superlattice." This technique prevents about 97 percent of "threading dislocations"—dislocations that occur at the junction of the first gallium arsenide layer and the silicon base—and almost all other dislocations as well.

According to Morkoc, one advantage of using the composite material is that current silicon fabrication equipment can be adapted rather easily to this hybrid technology.

Work at the University of Illinois on gallium arsenide/silicon semiconductor materials and devices is supported by the US Air Force Office of Scientific Research.

Nominations sought for McDowell Award

The chair of the W. Wallace McDowell Awards Committee has requested nominations for the 1986 award. Established in 1976 through an IBM grant, the McDowell Award is the Computer Society's highest technical achievement award. It is presented annually to a person whose professional work has been outstanding in concepts, technology, programming, education, or management in the computer field.

Nominations must include the name and address of the nominee, the names and addresses of three persons familiar with the nominee's work, a proposed citation, and at least five paper, book, or patent references to the work for which the nominee is to be recognized.

Send nominations by October 15 to John Bauer, Chair, W. Wallace Award, IEEE Computer Society, 10662 Los Vaqueros Circle, Los Alamitos, CA 90720.

NSF-ICOT establish cooperative program

The National Science Foundation and the Japanese Institute for New Generation Computer Technology are initiating a cooperative program to support US computer scientists and engineers performing research at ICOT.

Under the agreement, ICOT will begin receiving up to three NFS-selected visitors in 1987. Support will be provided by both organizations for stays averaging six months to one year. A speaking knowledge of Japanese, although desirable, is not required of applicants.

Proposals for support will be accepted at any time. They should include a description of the proposer's research interests, vita, and budget. Questions may be directed to Y.T. Chien or Kent K. Curtis, Division of Computer Research, National Science Foundation, 1800 G Street, NW, Washington, DC 20550.

COPP urges support for Information Age Commission Act

On August 29, the Computer Society's Committee on Public Policy subcommittee chairpersons met in Phoenix, Arizona, to discuss plans for 1986/87.

They were greeted by the news that Senate Bill S.786, the Information Age Commission Act, had just come out of committee. The act proposes to establish a two-year commission to define the impact of computers and communications on our nation and to identify ways of maximizing the technology's benefits to society.

The Committee on Public Policy endorsed this act in July 1985, and its members wrote letters to Senator Lautenberg (D-NJ) offering their support. COPP now urges all Computer Society members who are US citizens to familiarize themselves with this act and, if they agree that it's needed, to inform their senators and representatives. The committee's position, according to COPP Chair Ned Kornfield, is that the time has come to plan for the uniform impact of the computer on society, instead of letting it haphazardly grow in bits and pieces.

Other issues. Another item for COPP discussion is the stand the Computer Society should take regarding the professional engineering examination. With public safety becoming more dependent on computer controls, the adequacy of the PE exam and of one-time certification deserves investigation. Certification—and thus authorization to sign off on safety problems as different as a nuclear power plant cooling system and a water-cooled computer system, a building utilities system, or a waste disposal system—is now based on at most four (out of 24) computer-related questions.

COPP is also considering a position paper regarding the desirability for a nation-wide emergency computer communications network, similar to that of the Federal Emergency Management Agency, in conjunction with the American Radio Relay League. "Ham" computer people, tied through cellular telephone modems, could help locate the resources needed in an emergency with a minimal loss of time. This would be a citizens' version of the 911 emergency phone number.

COPP also sees a need for establishing a national policy for maintaining the privacy and security of computer-stored and network-transmitted information—for example, medical, banking, and employment information. The subcommittee's research indicates that most employers, banks, and hospitals are unaware of the inadequacy of the measures they use to safeguard the electronically-stored information, and also of its volatility.

COPP objectives. The overall objective of the Committee on Public Policy is to develop positions on those public policy issues that are of interest to the Computer Society, its membership, and the IEEE. After several years of discussion with a relatively small group of interested and dedicated people, the committee is now expanding opportunities for additional volunteer membership participation through its subcommittee structure (see box).

Membership. Interested Computer Society members can join in COPP subcommittee activities as either participating or corresponding members.

Participating membership entails active contributions to those issues confronting the subcommittee. Members in this category are expected to attend roughly two-thirds of the scheduled subcommittee meetings, take an active role in defining public policy issues, and help formulate appropriate position documentation.

Considerably less attendance and participation is expected of participating members. Duties may amount to as little

Computer Society Committee on Public Policy subcommittees, scopes, and chairpersons

The Committee on Public Policy has nine subcommittees, each focusing on a particular area of concern as defined below. Persons interested in participating are urged to contact the chairperson.

Beginning with the November issue, *Computer* will publish a regular column on COPP activities. Leading off will be a report from Egon E. Loebner, who chairs the Public Information Access Subcommittee, entitled "Whither Computer Technology: Populist or Centrist?"

Education/Computer Literacy. Scope: To identify, evaluate, and disseminate information regarding education and computer literacy issues as applied to the computer industry, including computer-based tools; special emphasis is placed upon impact studies and US legislative activity; to foster a close working relationship within the Computer Society, IEEE USAB, CSAB, ABET, and other related IEEE entities; to develop positions regarding issues of significance to the Computer Society.

Chair: Gary Raymond, Info Pro, Suite 3085, 2625 Butterfield Rd., Oak Brook, IL 60521; (312) 887-7880.

Human Resources/Immigration. Scope: To identify, evaluate, and disseminate information regarding human resource requirements within the computer profession and the impact of foreign nationals on these requirements with special emphasis placed upon US legislative activities in this area; to formulate positions for consideration by the Computer Society; to work closely with IEEE USAB regarding such positions.

Chair: Lewis Unnewehr, Allied Auto Tech Center, 900 W. Maple Rd., Troy, MI 48084; (313) 362-8454.

Information Transfer Policy. Scope: To identify, evaluate, and disseminate information regarding national policies which restrict information flow within the US through legislation, classification, and Presidential order; to develop guidelines by which the Computer Society membership can function when any restrictions are imposed; to develop positions regarding issues of significance to the Computer Society; to work closely with IEEE USAB and other comparable organizations regarding such positions.

Chair: Marshall Yovitz, Purdue University, 1125 E. 38th St., Indianapolis, IN 46205; (317) 923-1321.

Intellectual Property Protection. Scope: To identify, evaluate, and disseminate information regarding intellectual property protection issues as applied to computer engineers and computer scientists with special emphasis placed upon impact studies, US legislative activity, and extensions of copyright laws; to foster a close working relationship within the Computer Society, IEEE USAB, and other IEEE related activities; to develop positions regarding issues of significance to the Computer Society membership.

Chair: Richard H. Stern, 2101 L Street, NW, Suite 800, Washington, DC 20037; (202) 775-4727.

Privacy/Security. Scope: To identify, evaluate, and disseminate information regarding privacy and security issues as applied to all phases of the computer industry with special emphasis placed upon impact studies and US legislative activity; to foster a close working relationship within the Computer Society, IEEE USAB, and other IEEE related activities; to develop positions regarding issues of significance to the Computer Society and its members.

Chair: Jerry P. Jennings, Laddis Corp., PO Box 2008, Dearborn, MI 48123; (313) 525-9251.

Public Information Access. Scope: To foster the creation and introduction of means to provide the public with affordable computer-aided, interactive information access to administrative, legislative, and similar functions of governmental and other public activity bodies.

Chair: Egon E. Loebner, Hewlett-Packard Laboratories, 1501 Page Mill Rd., Palo Alto, CA 94304-1181; (415) 857-8787.

Social Issues. Scope. To identify, evaluate, and disseminate information on the impact ofcomputer technology on the quality of life and the consequences of these applications in the US and elsewhere; to explore ways that computers can be used in the future to improve the quality of life further, using the knowledge already acquired, and to determine the types of information that public officials need to set policies; to foster a close working relationship within the Computer Society, IEEE USAB, and other IEEE related activities.

Chair: Kenneth R. Anderson, Siemens Research & Technology, Princeton Forrestal Center, 105 College Rd. East, Princeton, NJ 08540; (609) 734-6550.

Technical Expert Directory. Scope: To identify technical experts in the field of computers, computer systems, their application and social implications, who are willing to testify before congressional committees as technical experts or to provide technical status reports; to collect names, addresses, telephone numbers, and previous statements of positions of these experts; to maintain a current database of that information; to foster a close working relationship within the Computer Society, IEEE USAB, and other IEEE related activities.

Chair: Claud M. Davis, IBM Corp., TPD Rm2B75, 500 Columbus Ave., Thornwood, NY 10594; (914) 742-5929.

Transnational Information Interchange. Scope: In cooperation with other professional bodies, to encourage to the maximum extent possible the interchange of computer-related technical information between the US and other nations through publications, technical meetings, and informal discussions; to encourage visits and technical personnel interchanges between nations; to foster a close working relationship with other technical organizations of international scope.

Chair: Sam Daram, 167 Rinconada Ave., Palo Alto, CA 94301; (415) 321-9095.

as responding to appropriate correspondence with suggestions, comments, and critiques via telephone, mail, or Compmail.

The committee is currently updating its mailing list. Prospective members or current members who wish to change their membership status should contact Paul Davis, Secretary, IEEE Computer Society Committee on Public Policy, 41 W. Huron St., Jackson, OH 45640; (614) 289-2331 x5794. Inquiries on a specific subcommittee's activities may be directed to the subcommittee chairperson.

Erratum

In the September issue of *Computer*, some equations in the article "A Proposed Standard Format for RSA Cryptosystems" by Philip Zimmermann were inadvertently printed with errors. The equations in the center column of page 23 should read:

$p_2 = ((C \bmod p)^{d \bmod (p-1)}) \bmod p$
$q_2 = ((C \bmod q)^{d \bmod (q-1)}) \bmod q$
If $p_2 = q_2$, then $M = p_2$, and the algorithm halts. Otherwise,
$p_2 = p_2 - (q_2 \bmod \mathrm{p})$
If $p_2 < 0$, then $p_2 = p_2 + p$
$M = q_2 + (q * ((p_2 * u) \bmod p))$

Hardware advances reflected in World Computer Chess Championship

by Monroe Newborn
McGill University, Montreal

If a program didn't run on more than one computer, or if a program didn't use special-purpose hardware, then it didn't finish in the top four places in the recent World Computer Chess Championship held in Cologne, Germany.

Cray Blitz, the winner of the five-round Swiss-style tournament ran on two processors of a Cray 2 for the first three rounds and on all four processors of a Cray XMP for the last two rounds. Executing 420 million instructions per second, Robert Hyatt, Burt Gower, and Harry Nelson's Cray Blitz examined trees having as many as 25 million nodes while deciding upon a move.

The second-place finisher, Hitech, ran on a Sun workstation with a specially designed VLSI move generator and position scorer attached. It, too, examined very large trees and perhaps played the best chess of the tournament, but a last-round loss to the winner cost the program the title. The program, developed at Carnegie Mellon University by a team headed by Hans Berliner, has competed in several tournaments with strong humans this year and has received performance ratings just short of a Grandmaster!

The third-place finisher, Bebe, developed by Tony Scherzer of Sys-10, Hoffman Estates, Illinois, also used a specially designed move generator and position scorer.

The fourth-place finisher, Jonathan Schaeffer's Phoenix, broke the previous record of eight computers searching together as a team (held by Ostrich, which was developed by this reporter) by competing with 20 Sun workstations which searched the chess trees in parallel.

All four programs actually finished tied with four out of five points, but the tie-breaking rules led to the ordering indicated.

Twenty three programs competed in the championship, five from the USA, four each from West Germany, England, and The Netherlands, three from Canada, and one each from Hungary, Denmark, and Sweden. Six programs were running on computers in North America and connected via satellite to the tournament hall.

Need Customized Systems Software?

For development tools and runtime libraries tailored to your special needs, contact Pecan Software Systems' Custom Software Division.

Pecan is the home of UCSD Pascal™ and the Power System™ family of program development aids. Our experienced systems programmers are prepared to mold any of our compilers or development tools to suit your specific requirements.

So don't settle for stock if your job demands the unique. Call us. You'll find that we can give you exactly what you need, when you need it. At a price you can afford.

- *UCSD Pascal*
- *Module-2*
- *FORTRAN-77*
- *BASIC*
- *Assemblers*
- *Cross Assemblers*
- *Debuggers*
- *Device Drivers*
- *Math Libraries*
- *Text Editors*
- *File Handling*
- *Multitasking*
- *Graphics*
- *Communications*
- *Database*
- *Windowing*
- *Networking*
- *Compiler Construction*
- *Operating Systems*

PECAN

Pecan Software Systems, Inc.
1410 39th Street, Brooklyn, N.Y. 11218
718-851-3100

Reader Service Number 3

Order #581

Forty-five papers organized into five sections in which the designs and applications of various supercomputers are discussed. The tutorial covers the following aspects of supercomputers: systems architecture, technology bases, large-scale computations, language extensions, vector processing, compiling techniques, commercial and exploratory systems, parallel algorithms, resource allocation, important applications, dataflow & VLSI computing, and future trends in supercomputing. 640 pp.

Tutorial Supercomputers; Design and Applications
Nonmembers—$36.00 Members—$24.00
Handling Charges Extra
Order from IEEE Computer Society Order Dept.
PO Box 80452, Worldway Postal Center
Los Angeles, CA 90080 USA
(714) 821-8380

"Any clod can have the facts, but having opinions is an art."
Charles McCabe,
San Francisco Chronicle

THE OPEN CHANNEL

The emperor's new angels

It started with noble goals and high hopes. "Wouldn't it be wonderful," the Emperor said, "if pigs had Wings? We could dispense with the Blue Angels, and banish forever the Danger To Mankind of close-formation flying."

So the Supersonic Pig Imperative was launched and the SPI Outfit was given a few billion Bucks for Exploratory Research. A Blue Ribbon Committee was assembled to address feasibility and cost. Its Report estimated that a Mach 2.71828 squadron of Pig Angels could be deployed in 3.6525 kilodays for 1.2345 times the national debt. A few technical Breakthroughs would be needed. Each Problem should have its own Panel, to schedule Breakthroughs and to recommend Research Funding.

A few biologists warned that pigs wouldn't grow Wings any time in that century. But the SPI had the Emperor's Own Blessing and the Prospect of Unlimited Funding. So Experts were found to serve on all the Panels, Contractors to take all the Contracts, Researchers to work on all the Problems.

The Propulsion System Panel assumed that the Wings Problem would be solved somehow, and spent its time studying high-flatulence foods. The Porcine Aerodynamics Panel pointed out the need for new wind tunnels and high-speed computers, regardless of what was decided about Wings. The Wing Management Panel started work on its longest lead-time item: Stratosphere Qualification of pigskin.

After five years, the Expert Piloting Panel noticed that the pigs in the flight simulators weren't very good close-formation pilots. The EPP kept adding Rules, but the pigs just didn't get any better. The SPI Outfit provided more Funding for Long-Term Research; as an Interim Measure, they decided to equip each pig with a human pilot.

Five years after that (just weeks before the scheduled Imperial Debut of the Pig Angels) the Wing Management Panel acknowledged that none of the Contractors had yet produced a flying pig. The Emperor kept drawing Wings with Crayons, but no one could make them stick. A Supplementary Appropriation was needed for more Research in Artificial Wings, Genetic Engineering, and Behavioral Psychology. Strictly as an Interim Measure, the SPI Outfit decided to place both pig and pilot in a high-performance aircraft.

After another five years, the Pig Angels were finally deployed in specially designed wide-body planes with cockpits large enough to accommodate the pigs and their pilots. The Pig Angels crashed a lot more often than the Blue Angels ever had, and their formation flying was never as good, but by then Everyone Who Counted was living too high off the hog to notice.

Moral: The impossible always takes a little longer and costs a lot more.

Jim Horning
DEC Systems Research Center
Palo Alto, California

The Open Channel is exactly what the name implies: a forum for the free exchange of technical ideas. Try to hold your contribution to one page maximum in the final magazine format (about 1000 words).

We'll accept anything (short of libel or obscenity) so long as it's submitted by a member of the Computer Society. If it's really bizarre we may require you to get another member to cosponsor your item.

Send everything to **Jim Haynes, Computer Center, UC Santa Cruz, CA 95065. Bitnet: haynes@ucscc; Arpa: ucscc!haynes@ucbvax.berkeley.edu.**

Which system is a RISC?

Recently, we've witnessed the appearance of several commercial systems that are either self-proclaimed RISC-type machines or are declared RISCs by others. We've also heard some finger-pointing statements: "Machine X is not really a RISC" and so on.

This correspondence proposes an orderly procedure to determine, at least loosely, which systems qualify as reduced instruction set computers and deserve the RISC label.

First of all, based on existing and well-recognized RISC systems, let's accept the following eight points of characterization:

(1) Number of machine-language instructions—as small as possible.

(2) Number of addressing modes—as small as possible.

(3) Number of instruction formats—as small as possible.

(4) Single cycle execution of all or most instructions.

(5) Memory access by load/store instructions only.

(6) Number of user's general-purpose CPU registers—as large as possible.

(7) Hardwired control.

(8) Support of high-level languages.

These points should not be regarded as a strict definition; a system would not have to satisfy all of them to be characterized as a RISC. Let's accept satisfaction of five of the eight points as sufficient for a system to be called a RISC-type machine. Naturally, one could argue that some of the points, such as (6), have nothing to do with the RISC idea, but arguments of this nature will never stop.

Twelve actual systems are taken as examples in the following considerations. Four are experimental prototypes:

- Berkeley RISC II,
- Stanford MIPS,
- IBM 801
- George Mason University Miris (Microcoded RISC)

The other eight are commercial systems:

- Pyramid
- Ridge

- Acorn (UK)
- INMOS transputer (UK)
- IBM RT PC
- Hewlett Packard HP3000/930
- Metaforth MF1600 (UK)
- Fairchild Clipper

While points (1), (2), (3), and (6) of the RISC characterization can be specified by numerical values, points (4), (5), (7), and (8) can be specified by a Yes(Y) or No(N) answer. (See Table 1.)

Let's now propose a loose definition of satisfaction or violation of the eight points. Points (4), (5), (7), and (8) are obviously satisfied by a Yes answer. For the rest, let's assume satisfaction for the following values:

(1) Number of instructions ≤ 150.

(2) Number of address modes ≤ 4.

(3) Number of instruction formats ≤ 4.

(6) Number of user CPU general registers ≥ 32.

Using S for "condition satisfied" and V for "condition violated," the information now appears as Table 2.

As we can see, none of the machines has more than three violations. Therefore, it is indeed a pleasure to declare that all of them are RISC-type systems.

Daniel Tabak
George Mason University
Fairfax, Virginia

Table 1. RISC characterization.

	Property							
System	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
RISC II	39	2	2	Y	Y	138	Y	Y
MIPS	31	2	4	Y	Y	16	Y	Y
801	120	3	2	Y	Y	32	Y	Y
Miris	64	2	4	Y	Y	2048	N	Y
Pyramid	128	16	3	N	Y	528	N	Y
Ridge	128	2	3	N	Y	16	N	Y
Acorn	44	2	6	Y	Y	16	Y	Y
INMOS	111	1	1	Y	Y	6	N	Y
RT PC	118	2	2	N	Y	16	N	Y
HP	140	2	1	Y	Y	32	Y	Y
MF1600	70	0	0	N	Y	1024	Y	Y
Clipper	101	9	14	Y	Y	16	Y	Y

Table 2. RISC satisfaction/violation.

	Property							
System	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
RISC II	S	S	S	S	S	S	S	S
MIPS	S	S	S	S	S	V	S	S
801	S	S	S	S	S	S	S	S
Miris	S	S	S	S	S	S	V	S
Pyramid	S	V	S	V	S	S	V	S
Ridge	S	S	S	V	S	V	V	S
Acorn	S	S	V	S	S	V	S	S
INMOS	S	S	S	S	S	V	V	S
RT PC	S	S	S	V	S	V	V	S
HP	S	S	S	S	S	S	S	S
MF1600	S	S	S	V	S	S	S	S
Clipper	S	V	V	S	S	V	S	S

Analytic Systems/Intelligence Information Processing

CIA . . .where your

CREATIVITY

turns data into intelligence.

At the Central Intelligence Agency, the heart of our operations is the analysis of raw information and its development into finished intelligence for the President and other senior U.S. policymakers. Many of the analytic tools for this critical function are developed by the Office of Research and Development.

To make important contributions to our analytic systems and intelligence information processing, we are seeking highly creative professionals with expert analytic skills, a strong background in Mathematics, Operations Research, or Computer Science, and interests in:

- Initiating and directing multidisciplinary research in analytic methodologies
- Applying operations research, mathematics, and computer science techniques to intelligence information processing and modeling
- Developing analytic models in the form of:
  - —simulation systems
  - —expert systems
  - —advanced graphics
  - —advanced computer applications

These positions require a PhD, or equivalent, and a minimum of two years experience. U.S. citizenship (both self and spouse) is mandatory. Employment is contingent upon full security and medical clearance.

Positions are located in the Washington, D.C. metropolitan area; some travel is required. The salary range is $30,000 to $50,000 commensurate with education and experience.

For further information, send a detailed resume and request for application form to:

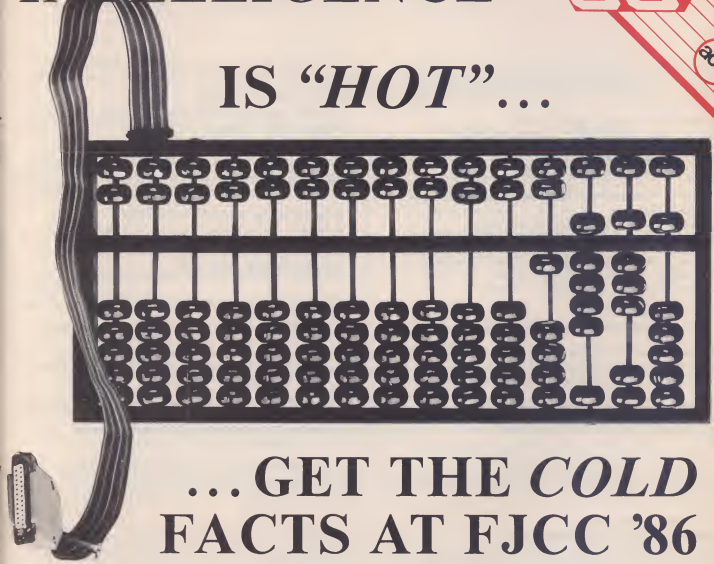
Recruitment Officer (Y09)
Office of Research & Development/ACS
Central Intelligence Agency
Washington, D.C. 20505

Central Intelligence Agency

The CIA is an equal opportunity employer Written inquiries only. We will respond within 30 days to those judged to be of further interest

ARTIFICIAL INTELLIGENCE

IS *"HOT"*...



...GET THE *COLD* FACTS AT FJCC '86

SEE NEXT PAGE ▶

REGISTER: Call 1-800-722-FJCC

FALL JOINT COMPUTER CONFERENCE—2-6 NOVEMBER 1986

© 1986 Worldwide Data Communications Co.

FALL JOINT COMPUTER CONFERENCE—2-6 NOVEMBER 1986

Artificial Intelligence (AI)

13 TECHNICAL SESSIONS

AI-1: Artificial Intelligence Technology
Chair: Dr. Elaine Kant, Schlumberger-Doll Research Center
Session 1: "Design Issues and Practice in AI Programming" (Rm. 2-H)
Tuesday, November 4, 1:30 pm–3:30 pm
Session Chair: *Nancy Martin, Softpert Systems Inc.*

AI-2: Computer Vision
Chair: Prof. John Kender, Columbia U.

Session 1: "Computational Approaches to Low and Middle-Level Vision" (Rm. 4-I)
Wednesday, November 5, 10:00 am–12:00 noon
Session Chair: *Prof. John Kender, Columbia U.*

Session 2: "Model-Based High-Level Vision" (Rm. 5-I)
Wednesday, November 5, 1:30 pm–3:30 pm
Session Chair: *Prof. John Kender, Columbia U.*

AI-3: Robotics
Chair: Prof. Richard Paul, University of Pennsylvania

Session 1: "Robot Perception" (Rm. 6-I)
Wednesday, November 5, 3:45 pm–5:15 pm
Session Chair: *Prof. T. Kanade, Carnegie Mellon U.*

Session 2: "Task-Level Robot Programming" (Rm. 7-G)
Thursday, November 6, 10:00 am–12:00 noon
Session Chair: *Prof. Tomas Lozano-Perez, MIT*

Session 3: "Real-Time Robot Programming" (Rm. 8-G)
Thursday, November 6, 1:30 pm–3:30 pm
Session Chair: *Dr. Russell Taylor, IBM T.J. Watson Research Center*

AI-4: Rule-Based Systems
Chair: Prof. David Rine, George Mason U.

Session 1: "Software Engineering for Rule-Based Systems" (Rm. 5-C)
Wednesday, November 5, 1:30 pm–3:30 pm
Session Chair: *Prof. David Rine, George Mason U.*

Session 2: "Knowledge Engineering" (Rm. 6-C)
Wednesday, November 5, 3:45 pm–5:15 pm
Session Chair: *Dr. Richard Wexelblat, Philips Laboratories*

Session 3: "Rule-based Models and Applications" (Rm. 8-C)
Thursday, November 6, 1:30 pm–3:30 pm
Session Chair: *Dr. Elaine Kant, Schlumberger-Doll Research Center*

Session 4: "PROLOG and Frame-Based Methods" (Rm. 9-C)
Thursday, November 6, 3:45 pm–5:15 pm
Session Chair: *Dr. Thomas Bottegal, George Mason U.*

9 PROFESSIONAL EDUCATION COURSES

***Building Expert Systems Workshop*—Sun. # 390 or Mon. # 395**
Instructor: John D. McGregor, Dept. of Computer Studies, Murray State U.

***Knowledge-Based Expert Systems*—Sun. # 320 or Mon. # 325**
Instructors: Lois Boggess and Julia Hodges, Dept. of Computer Science, Mississippi State U.

***Architectures for AI Applications*—Sun. # 380 or Mon. # 385**
Instructor: Benjamin W. Wah, Dept. of Electrical and Computer Engineering, U. of Illinois at Urbana-Champaign.

***Natural Language Processing*—Sun. # 315**
Instructor: Michael Lebowitz, Dept. Computer Science, Columbia U.

***Machine Learning*—Mon. # 950**
Instructor: Michael Lebowitz, Dept. of Comp. Sci., Columbia U.

***Speech Recognition: From Isolated Digits to Natural Language Dictation*—Sun. # 830**
Instructor: Paul Bamberg, Prin. Scientist, Dragon Systems.

***Computer Vision from an AI Perspective*—Mon. # 940**
Instructors: John Kender and Takeo Kanade, Dept. Computer Science, Columbia U.

***PROLOG & Knowledge Info. Processing*—Sun. # 370 or Mon. # 375**
Instructor: Douglas DeGroot, V.P. (R&D) Quintus Computer Systems

****AI Programming and Environments*—Sun. and Mon. # 340**
Instructor: Harlan H. Black, Computer Scientist, U.S. Army, Communications/Electronics Command (ARES Project).

*Two day course

13 Technical Sessions (Continued)

AI-5: Natural Language Processing
Chair: Prof. Robert Wilensky, U. of California at Berkeley

Session 1: "User Interfaces" (Rm. 1-I)
Tuesday, November 4, 10:00 am–12:00 noon
Session Chair: *Prof. Robert Wilensky, U. of California at Berkeley*

Session 2: "Natural Language Processing" (Rm. 2-I)
Tuesday, November 4, 1:30 pm–3:30 pm
Session Chair: *Prof. Richard Granger, UC at Irvine*

Session 3: "Problems And Prospects Of NLP" (Rm. 3-I)
Tuesday, November 4, 3:45 pm–5:15 pm
Session Chair: *Dr. Phillip J. Hayes, Carnegie Group Inc.*

EXPLORE THE KNOWLEDGE-BASED SOCIETY AT THE FALL JOINT COMPUTER CONFERENCE

FALL JOINT COMPUTER CONFERENCE—2-6 NOVEMBER 1986

FJCC '86 CONFERENCE REGISTRATION FORM

Please send this registration form with payment to FJCC '86, INFOMART, 1950 Stemmons Freeway, Dallas, TX 75201: Phone: 1-800-722-FJCC
Advance registration discounts are available to all registration forms received, with full payment, post-marked on or before October 18.
Attendees who have pre-registered for the Conference or the Professional Education Program may pick up their conference credentials beginning on Saturday, November 1 at 12 Noon until 5 PM at INFOMART/Plaza Level. Please bring your registration acknowledgment with you to save time in line. Notices of cancellation must be received in writing at the FJCC Registration Office NO LATER than October 27, 1986 in order to qualify for a refund. Allow 10 weeks after the Conference for processing of refunds.

NAME (last) (first) (MI)

POSITION INSTITUTION

ADDRESS CITY STATE ZIP

PHONE MAIL LIST RESTRICTIONS: ☐ NO RESTRICTION ☐ PROFESSIONAL SOCIETY USE ONLY

TO QUALIFY FOR MEMBER OR STUDENT MEMBER DISCOUNT, ☐ ACM or ☐ IEEE-CS MEMBER NUMBER: ______

NOTE: CIRCLE FEE FOR REGISTRATION TYPE SELECTED

FJCC '86 CONFERENCE	ADVANCE REGISTRATION (Before Oct. 18, 1986)	REGULAR/ON-SITE REGISTRATION
FULL CONFERENCE & EXHIBITS		
Member	$150	$200
Non-Member	190	250
Student	25	35
(Includes Conference Proceedings)		
ONE DAY ONLY		
Member	90	120
Non-Member	120	150
(No Proceedings)		
Circle Day:	MON TUE WED THURS	
EXHIBITS ONLY (No Proceedings)	25	25

JOB FUNCTION

- ☐ Management 01
- ☐ Project Leader 02
- ☐ Systems Analyst 03
- ☐ Software Engineer 04
- ☐ Programmer 05
- ☐ Data Base Manager 06
- ☐ Mktg. Comm. Eng. 07
- ☐ Univ./Coll. Eng. 08
- ☐ Pri./Sec. Educator 09
- ☐ Research 10
- ☐ Administrator 11
- ☐ Consultant 12
- ☐ Other________ 13

INDUSTRY

- ☐ Banking/Fin. 14
- ☐ Comp. Svcs. Reseller 15
- ☐ Consulting 16
- ☐ DP Services 17
- ☐ Education 18
- ☐ Engr./Arch. Svcs. 19
- ☐ Gov't 20
- ☐ Info. Sys. Mfg. 21
- ☐ Med/Health Care 22
- ☐ Petrochem. 23
- ☐ Retail 24
- ☐ Transportation 25
- ☐ Press 26
- ☐ Speaker 27
- ☐ Other ________ 28

Professional Education Program Course	One-Day Courses		Two-Day Courses	
	Advance Registration (Before Oct. 18, 1986)	Regular/On-Site Registration	Advance Registration (Before Oct. 18, 1986)	Regular/On-Site Registration
Member	$200	$240	$450	$550
Non-Member	$250	$300	$550	$675
Course Number(s)	Sunday ☐☐☐	Monday ☐☐☐	Sunday and Monday ☐☐☐	

A check or money order made payable in US funds to FJCC '86, for the full amount of the registration must accompany your registration form. Purchase orders will be accepted. The following credit card authorization may also be used
Amount enclosed or authorized to charge my credit card: U.S. $________Card No.: ________

☐ VISA ☐ AMER. EXP. ☐ MC Exp. Date: ________ Signature________Date: ________
☐ I plan to attend the Complimentary Special Joint Conferences and Exhibitor's Reception on November 3, 1986 at INFOMART.

CS 10/86

FJCC '86 HOTEL RESERVATION FORM

Mail this form to: FJCC Housing Bureau, Dallas Convention Bureau, 1507 Pacific Avenue, Dallas, TX 75201 – Telephone: 214-954-1440. Room requests must be **received** by October 3, 1986.

To receive special "Conference Rates," at the hotels listed below, our staff must make your reservation. **Please indicate hotel choice by writing the numbers 1, 2, 3, and 4 next to the hotels as appropriate.** IMPORTANT: Phone requests for reservations will NOT be honored.

FJCC Housing Bureau will acknowledge receipt of your hotel reservation request. Confirmation of your hotel reservation will be sent directly from the hotel. Please review your hotel confirmation closely for specific deposit requirements of each hotel. **Deposits** for rooms should be sent directly to the hotel after your receive your confirmation, and are required if you plan on arriving after 6 PM. Deposits will be refunded by the hotel if you cancel at least 48 hours in advance of the arrival date. Hotels will not guarantee availability of your room beyond 6:00 PM unless a one night's deposit has been sent to the hotel. DO NOT SEND DEPOSIT TO THE FJCC HOUSING BUREAU.

Cancellations and changes in reservations should be made in writing to the FJCC Housing Bureau until October 3. After October 3, cancellations and changes should go directly to the hotel.

Choice	Hotel	☐ Single	☐ Double	☐ Triple	☐ Suite
	Loews Anatole	$90	$104	$120	Available Upon Request
	Marriott Market Center	$75	$ 85	$ 85	Available Upon Request
	Wyndham*	$85	$100	$115	N/A
	Viscount	$48	$ 52	$ 58	N/A

9% tax is applicable to all above rates. *Block of rooms available only from October 30th through November 4th.

NAME (last) (first) (MI)

POSITION INSTITUTION

STREET ADDRESS

CITY STATE ZIP

BUSINESS PHONE Check-in Date/Time: Check-out Date/Time:

CS 10/86

CALENDAR

October 1986

Micro 19, 19th Annual Workshop on Microprogramming (ACM), October 15-17, New York, New York. Contact S. Habib, CCNY, Computer Science Dept., New York, NY 10031; (212) 690-6631.

1986 ACM Sigcpr International Conference on New Careers in Computing, October 16-17, Calgary, Alberta, Canada. Contact Paul Licker, Faculty of Management, University of Calgary, Calgary, Alberta T2N 1N4, Canada.

Workshop on Design Principles for Experimental Distributed Systems, October 16-17, West Lafayette, Indiana. Contact Samuel D. Conte, Dept. of Computer Science, Computer Science Bldg., Purdue University, West Lafayette, IN 47907; (317) 494-6005.

Second Annual Conference on Expert Systems in Government, October 20-24, McLean, Virginia. Contact Kamal Karna, CCG, 823 Flagler Dr., Gaithersburg, MD 20878; (301) 921-0392.

Applied Imagery Pattern Recognition, October 22-24, Washington DC. Contact Dennis Lynch, PSC, Inc., 10340 Democracy Lane, Fairfax, VA 22030; (703) 385-6880.

Conference on Linking Technology and Users (IEEE), October 22-24, Charlotte, North Carolina. Contact Roger Grice, IBM Corp., 60L 865, Neighborhood Rd., Kingston, NY 12401; (914) 385-3675.

IPCC-86, International Professional Communication Conference (IEEE), October 22-24, Charlotte, North Carolina. Contact Lois K. Moore, Johns Hopkins University, Applied Physics Laboratory, Johns Hopkins Rd., Laurel, MD 20707; (301) 953-5000, ext. 8313 or Steve Doneny-Farina, University of North Carolina, Dept. of English, UNCC Station, Charlotte, NC 28223; (704) 597-4204.

1986 Workshop on Interactive 3D Graphics (ACM), October 22-24, Chapel Hill, North Carolina. Contact Mary Whitton, Trancept Systems, Inc., 521 Uwharrie Ct., Raleigh, NC 27606-1456; (919) 828-7406.

International Symposium on Methodologies for Intelligent Systems (ACM), October 23-25, Knoxville, Tennessee. Contact Zbigniew Ras or Maria Zemankova, The University of Tennessee at Knoxville, Computer Science Dept., Knoxville, TN 37996-1301; (615) 974-5067.

Workshop on Charged Coupled Devices (IEEE), October 24-26, Harriman, New York. Contact Eric R. Fossum, Dept. of Electrical Engineering, 1321 Mudd Bldg., Columbia University, New York, NY 10027; (212) 280-3115.

Medinfo 86, Fifth World Congress on Medical Informatics (AFIPS, AMA, IFIP, ASIS), October 26-30, Washington DC. Contact Secretariat, George Washington University, Medical Center, OCME, 2300 K St., NW, Washington DC 20037; (202) 676-8929.

Symposium on Foundations of Computer Science, October 27-29, Toronto, Canada. Contact Ashok Chandra, IBM T.J. Watson Research Center, PO Box 218, Yorktown Heights, NY 10598; (914) 945-1752.

32nd IEEE-Holm Conference on Electrical Contacts, October 27-29, Boston, Massachusetts. Contact Fern Katronetsky, IEEE Headquarters, 345 E. 47th St., New York, NY 10017-2394; (212) 705-7405.

Workshop on Large Grained Parallelism, October 27-29, Providence, Rhode Island. Contact Paul J. Leach, Apollo Computer, Inc., 330 Billerica Rd., Chelmsford, MA 01824; (617) 256-6600.

IEEE Computer Society International Conference on Computer Languages, October 27-30, Miami Beach, Florida. Contact Joseph Urban, University of S.W. Louisiana, Center for Advanced Computer Studies, PO Box 44330, Lafayette, LA 70504; (318) 231-6304.

Maecon 86, Mid-America Electronics Conference (IEEE), October 28-29, St. Louis, Missouri. Contact Robert E. Barba, PO Box 699, Utica, MI 48087; (313) 254-5310.

GaAs IC 86, 1986 IEEE Gallium Arsenide Integrated Circuits Symposium, October 28-30, Grenelefe, Florida. Contact Robert J. Markunas, Research Triangle Institute, PO Box 12194, Research Triangle Park, NC 27709; (919) 541-6153.

IEEE Workshop on Expert Systems and Pattern Analysis, October 31, Paris, France. Contact C. H. Chen, Electrical and Computer Engineering Dept., Southeastern Massachusetts University, North Dartmouth, MA 02747; (617) 999-8475.

November 1986

FJCC-86, 1986 ACM/IEEE-CS Fall Joint Computer Conference, November 2-6, Dallas, Texas. Contact IEEE Computer Society, 1730 Massachusetts Ave., NW, Washington DC 20036-1903; (202) 371-0101; TWX 7108250437 IEEECOMPSO.

1986 IEEE ASSP Workshop on VLSI Signal Processing, November 5-7, Los Angeles, California. Contact S. Y. Kung, Image Processing Institute, Powell Hall, University of Southern California, University Park, Los Angeles, CA 90007; (213) 743-6581.

1986 IEEE/Engineering in Medicine and Biology Society Eighth Annual Conference, November 7-10, Fort Worth, Texas. Contact George V. Kondraske, University of Texas at Arlington, Box 19138, Arlington, TX 76019; (817) 273-2335 or Charles J. Robinson, Rehabilitation R&D Center, Hines VA Hospital, Box 20, Hines, IL 60141; (312) 343-7200, ext. 2240.

ICCAD-86, IEEE International Conference on Computer-Aided Design (ACM), November 10-13, Santa Clara, California. Contact Ian E. Getreu, PO Box 1356, Beaverton, OR 97075; (503) 629-1462.

Comdex/Fall 86, November 10-14, Las Vegas, Nevada. Contact Linda M. Yogel or Anna Jew, The Interface Group, Inc., 300 First Ave., Needham, MA 02194; (617) 449-6600.

Autofact Conference and Exposition, November 11-14, Detroit, Michigan. Contact Society of Manufacturing Engineers, One SME Dr., PO Box 930, Dearborn, MI 48121; (313) 271-0023.

Latincon 86 (IEEE), November 11-14, Panama City, Panama. Contact Cristobal Silva, PO Box 10616, Panama 4, Panama; phone 62-1794; telex 2158 IRHEPA.

Conferences that the Computer Society participates in or sponsors are indicated by the IEEE Computer Society logo; other conferences of interest to our readers are also included. For inclusion in Calendar or Call for Papers, submit information **six weeks before the month of publication** (e.g., for the **January 1987** issue, send information for receipt by **November 15, 1986)** to COMPUTER, 10662 Los Vaqueros Circle, Los Alamitos, CA 90720.

Fifth Annual VLSI Packaging Workshop (IEEE), November 17-18, Paris, France. Contact Karel Kurzweil, BULL-Rue Jean Jaures, 78340 Les Clayes Sous Bois, France; phone (1) 34-62-70-48.

IEEE Computer Networking Symposium, November 17-18, Washington DC. Contact Tuncay Saydam, University of Delaware, Dept. of Computer and Information Sciences, 103 Smith Hall, Newark, DE 19716; (302) 451-2716.

Fifth International Conference on Entity-Relationship Approach (ACM, AFCET), November 17-19, Dijon, France. Contact Adarsh K. Arora, Gould Research Center, 40 Gould Center, Rolling Meadows, IL 60008; (312) 640-4712 or Stefano Spaccapietra, Universite de Bourgogne-IUT, BP 510—21014 Dijon Cedex, France; phone 80664611.

Third International Software Process Workshop (ACM), November 17-19, Breckenridge, Colorado. Contact Robert M. Balzer, Information Sciences Institute, 4676 Admiralty Way, Suite 1001, Marina del Rey, CA 90292; (213) 822-1511 or Mark Dowson, Imperial Software Technology, 60 Albert Ct., Prince Consort Rd., London SW7 2BH, UK. Workshop limited to 30 people.

Software Tools for AI/Expert Systems (ACM), November 24-25, Boston, Massachusetts. Contact Warren Briggs, School of Management, Suffolk University, 8 Ashburton Pl., Boston, MA 02108; (617) 723-2349.

December 1986

1986 Electrical and Electronics Conference and Exposition (IEEE), December 1-3, Toronto, Canada. Contact IEEE Canadian Region Office, 7061 Yonge St., Thornhill, Ontario L3T 2A6, Canada; phone (416) 881-1930.

Computer Elements Workshop, December 1-4, Phoenix, Arizona. Contact William Rosenbluth, IBM Corp. 866-C2F, 9500 Godwin Dr., Manassas, VA 22110; (703) 367-2408.

Globecom 86, 1986 Global Telecommunications Conference (IEEE), December 1-4, Houston, Texas. Contact Ross C. Anderson, Southwestern Bell, Room 706, 3100 Main St., Houston, TX 77002; (713) 521-8244.

Real-Time Systems Symposium, December 2-4, New Orleans, Louisiana. Contact John A. Stankovic, Dept. of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213; (412) 578-7668.

1986 ACM Conference on Personal and Small Computers, December 3-5, San Francisco, California. Contact Jacob Slonim, Geac Computers, 350 Steelcase Rd. West, Markham, Ontario L3R 1B3, Canada.

IEDM-86, 1986 International Electron Devices Meeting (IEEE), December 7-10, Los Angeles, California. Contact Melissa Widerkehr, Courtesy Associates, Inc., 655 15th St., NW, Washington DC 20005; (202) 347-5900.

WSC-86, 1986 Winter Simulation Conference (ACM, IIE, NBS, SCS), December 8-10, Washington DC. Contact James Henriksen, Wolverine Software Corp., 7630 Little River Turnpike, Annandale, VA 22003; (703) 750-3910.

Second ACM Sigsoft/Sigplan Symposium on Practical Software Development Environments, December 9-11, Palo Alto, California. Contact Peter B. Henderson, Dept. of Computer Science, SUNY at Stony Brook, Stony Brook, NY 11794; (516) 246-7145, ext. 7090.

1986 Decision and Control Conference (IEEE), December 10-12, Athens, Greece. Contact Anthony Ephremedes, Dept. of Electrical Engineering, 2443 Dept., University of Maryland, College Park, MD 20742; (301) 454-6871.

ICS-86, International Computer Symposium, December 15-19, Tainan, Taiwan, Republic of China. Contact Ming T. Liu, Dept. of Computer and Information Science, Ohio State University, 2036 Neil Ave., Columbus, OH 43210-1277; (614) 422-1837.

January 1987

20th Hawaii International Conference on System Sciences, January 6-9, Kailua-Kona, Hawaii. Contact Ralph H. Sprague, University of Hawaii, College of Business Administration, R-303, Honolulu, HI 96822; (808) 948-7430.

Symposium on Intelligent Control, January 18-21, Philadelphia, Pennsylvania. Contact A. Meystel, Drexel University, Dept. of Electrical and Computer Engineering, College of Engineering, Philadelphia, PA 19104; (215) 895-2241.

IOOFC-87, 1987 International Conference on Integrated Optics and Optical Fiber Communication (IEEE), January 19-22, Reno, Nevada. Contact OSA Meetings Dept., 1816 Jefferson Pl., NW, Washington DC 20036; (202) 223-0926.

OFC-87, 1987 Conference on Optical Fiber Communication (IEEE), January 19-22, Reno, Nevada. Contact OSA Meetings Dept., 1816 Jefferson Pl., NW, Washington DC 20036; (202) 223-0926.

Annual IEEE Design Automation Workshop, January 20-23, Apache Junction, Arizona. Contact James Armstrong, Dept. of Electrical Engineering, Virginia Tech, Blacksburg, VA 24061; (703) 961-7078.

14th ACM Sigact-Sigplan Symposium on Principles of Programming Languages, January 21-23, Munich, West Germany. Contact Steve Muchnick, Sun Microsystems, MS 5-40, 2550 Garcia Ave., Mountain View, CA 94043; (415) 960-7233 or Mark Wegman, IBM T. J. Watson Research Center, PO Box 218, Yorktown Heights, NY 10598; (914) 945-1327.

National Business Employment Weekly
Published by THE WALL STREET JOURNAL
HUNDREDS OF CAREER-ADVANCEMENT OPPORTUNITIES

Which one of our engineering positions is right for you?

Find out in the National Business Employment Weekly.

Every week, the National Business Employment Weekly, published by The Wall Street Journal, contains hundreds of the best executive, managerial and professional positions from all across the country. Top jobs in every field, including career opportunities in virtually every engineering discipline. At salaries from $25,000 to $250,000.

PLUS...weekly editorial features covering every aspect of career advancement. Articles on writing resumes, interviewing, salary statistics, regional employment trends, and much more.

LOOK FOR OUR SPECIAL NOVEMBER 2ND "ENGINEERING" ISSUE WHICH WILL INCLUDE ADDITIONAL CAREER OPPORTUNITIES FOR ENGINEERS AND RELATED EDITORIAL.

Pick up a copy of the National Business Employment Weekly at your newsstand today. Or we'll send you the next eight issues by first class mail. Just send a check for $35 to:

National Business Employment Weekly,
Dept. IC2, 420 Lexington Avenue,
New York, NY 10170.

Reader Service Number 4

CALL FOR PAPERS

Computer: Articles that cover the state of the art and important new developments in computer science, technology, and applications are sought. An authors' information sheet can be obtained from the editor-in-chief, Mike Mulder, who can be reached at the Applied Research Center, University of Portland, Portland, OR 97203; (503) 283-7433.

International Journal for Artificial Intelligence in Engineering: This journal emphasizes research and development leading to problem solving. Write for submission requirements to D. Sriram, Dept. of Civil Engineering, Carnegie Mellon University, Pittsburgh, PA 15213 or K. J. MacCallum, Dept. of Ship and Marine Technology, Marine Technology Centre, 100 Montrose St., Glasgow, UK.

Second Israel Conference on Computer Systems and Software Engineering (IEEE): May 6-7, 1987, Tel Aviv, Israel. Submit four copies of extended abstracts (five to 10 pages) to Jonah Z. Lavi, c/o Ortra, Ltd., POB 50432, Tel Aviv 61500, Israel; phone 972-3-971-3716.

Second International Conference on Production Systems (INRIA): April 6-10, 1987, Paris, France. Submit four copies of full paper (10 to 15 pages) including abstract (200 words maximum) by **October 10, 1986,** to INRIA, Service des Relations Exterieures, Bureau des Colloques, Domaine de Voluceau, Rocquencourt, BP 105, 78153 Le Chesnay Cedex, France.

1987 ASME International Computers in Engineering Conference and Exhibition: August 9-13, 1987, New York, New York. Papers are solicited in all areas related to the development, application, and research of computers in mechanical engineering. Contributions in the form of a full-length paper or an extended abstract are solicited. Submit abstracts by **October 15, 1986,** and submit three copies of finished contributions (papers or extended abstracts) by **November 30, 1986,** to T. James Cokonis, General Electric Co., Advanced Development and Information Systems, PO Box 8555, Bldg.7-Rm. 7244, Philadelphia, PA 19101; (215) 354-5112.

Second National Conference on Computer Applications: February 25-27, 1987, Lagos, Nigeria. Submit four copies of paper by **October 31, 1986,** to one of the following: Abimbola Salako, Computer Centre, University of Lagos, Akoka, Lagos, Nigeria; Josephine Guidi Wandja, Universite Nationale, 08 BP 217, Abidjan 07, Ivory Coast; Alfs T. Berztiss, Computer Science Dept., University of Pittsburgh, Pittsburgh, PA 15260; or Fabrizio M. Ferrara, Information Systems Division, Gestione Sistemi per l'Informatica S.R.L., Via Rodi 32, 00195 Rome, Italy.

37th Electronic Components Conference (IEEE): May 11-13, 1987, Boston, Massachusetts. Submit 10 copies of abstract and outline by **October 31, 1986,** to Doug Loescher, Sandia National Laboratories, Division 2123, PO Box 5800, Albuquerque, NM 87185.

Annual IEEE Design Automation Workshop: January 21-23, 1987, Apache Junction, Arizona. Attendance is limited to 55 persons. To participate, submit a short summary of interests and activities; to make a presentation, submit a short summary of proposed talk. Send both by **November 1, 1986,** to Jim Armstrong, Electrical Engineering Dept., Virginia Tech, Blacksburg, VA 24061; (703) 961-4723 or (703) 961-7078.

CIPS Congress 87: Canadian Information Processing Society Congress on Technology for People: May 12-14, 1987, Winnipeg, Manitoba, Canada. Submit four copies of an extended abstract (500 words maximum) by **November 1, 1986,** and four copies of the manuscript (20 pages maximum) by **December 1, 1986,** to M. S. Doyle, Computer Science, University of Manitoba, Winnipeg, Manitoba R3T 2N2, Canada; phone (204) 474-8456.

1987 International Symposium on Multiple-Valued Logic: May 26-28, 1987, Boston, Massachusetts. Submit four copies of a paper (20 pages maximum) that includes an abstract (50 to 100 words) by **November 1, 1986,** to Ivo Rosenberg, Research Center for Applied Mathematics, University of Montreal, Montreal, CP 6128, Succ. A, PQ H3C 3J7, Canada (authors in the Americas); H. G. Kerkoff, Twente University of Technology, Dept. of Electrical Engineering, IC-Technology and Electronics Group, 7500 AE Enschede, The Netherlands (authors in Europe/Africa); or Okihiko Ishizuka, Dept. of Electrical Engineering, Miyazaki University, 1-1-1 Kirishima, Miyazaki-shi 880, Japan (authors in Asia/Pacific).

NECC-87, Eighth National Educational Computing Conference: June 24-26, 1987, Philadelphia, Pennsylvania. Full-length papers, suggestions for tutorial sessions and panel discussions, and project presentations are sought. Submit four copies of the full-length paper (15 pages maximum) and a one-page abstract; submit one copy of suggestions for panel and tutorial sessions, both to NECC-87/JA1, International Council for Computers in Education, University of Oregon, 1787 Agate St., Eugene, OR 97403-1923. Send three copies of proposals for project presentations to NECC-87/JA1, Mary Beschorner, University of San Diego, School of Education, San Diego, CA 92110. **November 1, 1986,** is the deadline for submissions in all three categories.

Fifth International Symposium on Data Analysis and Informatics (INRIA): September 29-October 2, 1987, Versailles, France. Submit four copies of summary (500 words maximum) by **November 5, 1986,** to INRIA, Service des Relations Exterieures, Domaine de Voluceau, Rocquencourt, BP 105, 78153 Le Chesnay Cedex, France.

International Conference on Human-Computer Interaction: August 10-15, 1987, Honolulu, Hawaii. Submit 300-word abstract by **November 15, 1986,** to Gavriel Salvendy, School of Industrial Engineering, Purdue University, West Lafayette, IN 47907.

First International Conference on Industrial and Applied Mathematics (SIAM): June 29-July 3, 1987, Paris, France. Submit abstract by **November 28, 1986,** to SIAM, 117 S. 17th St., Philadelphia, PA 19103 or INRIA, Domaine de Voluceau, Rocquencourt BP 78153, Le Chesnay Cedex, France.

Sigplan 87, Symposium on Interpreters and Interpretive Techniques (ACM): June 24-26, 1987, St. Paul, Minnesota. Submit 11 copies of an 8-to-10-page summary (including an abstract of 250 words or less) by **November 28, 1986,** to Thomas N. Turba, Sperry Computer Systems, PO Box 64942, St. Paul, MN 55164.

IEEE Transactions on Computers: Papers are sought for two special issues planned for 1987. The first will be devoted to real-time systems. Submit six copies of a paper by **December 1, 1986,** to Kang G. Shin, Dept. of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109; (313) 763-0391. The second will be devoted to supercomputing. Submit six copies of a paper by **February 1, 1987,** to H. C. Torng, School of Electrical Engineering, Cornell University, Ithaca, NY 14853-5401; (607) 255-5191. (Guidelines for submitting manuscripts appear on the back cover of every issue of *IEEE Transactions on Computers.*)

Conferences that the Computer Society participates in or sponsors are indicated by the IEEE Computer Society logo; other conferences of interest to our readers are also included. For inclusion in Call for Papers or Calendar, submit information **six weeks before the month of publication** (e.g., for the **January 1987** issue, send information for receipt by **November 15, 1986**) to COMPUTER, 10662 Los Vaqueros Circle, Los Alamitos, CA 90720.

MFOC-87, First International Military Fiber Optic and Communications Exposition: March 16-19, 1987, Washington DC. Submit abstracts (100 words minimum) by **December 1, 1986,** to Danae Fasano, Information Gatekeepers, Inc., 214 Harvard Ave., Boston, MA 02134; (617) 232-3111.

ICCV-87, First International Conference on Computer Vision: June 8-11, 1987, London, England. Send four copies of complete paper by **December 15, 1986,** to Azriel Rosenfeld, Center for Automation Research, University of Maryland, College Park, MD 20742.

RTA-87, Second International Conference on Rewriting Techniques and Applications: May 25-27, 1987, Bordeaux, France. Submit 11 copies of one-page abstract and four copies of full paper (15 pages maximum) by **December 15, 1986,** to Pierre Lescanne, RTA-87, Centre de Recherche en Informatique de Nancy, Campus Scientifique, BP 239, 54506 Vandoeuvre-les-Nancy Cedex, France (authors in Europe) or David Plaisted, RTA-87, Dept. of Computer Science, New West Hall, 035-A, University of North Carolina at Chapel Hill, Chapel Hill, NC 27514 (authors in other areas).

AISB-87 Conference: April 6-10, 1987, Edinburgh, Scotland. Submit four copies of a paper (8000 words maximum) on any subject relevant to artificial intelligence and the simulation of behavior by **January 1, 1987,** to Chris Mellish, Arts Bldg., University of Sussex, Falmer, Brighton BN1 9QN, UK.

IJCAI-87, 10th International Joint Conference on Artificial Intelligence: August 23-28, 1987, Milan, Italy. Submit six copies of a paper (5600 words for full-length papers and 2400 words for short papers) and a 100-to-200-word abstract by **January 5, 1987,** to John McDermott, Dept. of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213.

First Conference of the International Federation of Classification Societies: June 29-July 1, 1987, Aachen, West Germany. To be considered for making a 25-minute presentation, submit a title and one-page abstract by **January 15, 1987,** to H. H. Bock, Institut fur Statistik und Wirtschaftsmathematik, Technical University Aachen, Wuellnerstrasse 3, D-5100 Aachen, West Germany.

1987 International Conference on Chinese and Oriental Language Computing: June 14-16, 1987, Chicago, Illinois. Submit five copies of extended summary (approximately 500 words) by **January 15, 1987,** to A. C. Liu, Dept. of Electrical and Computer Engineering, Illinois Institute of Technology, IIT Center, Chicago, IL 60616 (authors in the US); or Patrick S. P. Wang, College of Computer Science, Northeastern University, 360 Huntington Ave., Boston, MA 02115 (authors outside the US).

Second International Workshop on Applied Mathematics and Performance/Reliability Models of Computer/Communication Systems (IFIP, SCS**):** May 25-29, 1987, Rome, Italy. Submit five copies of a paper (15 pages maximum) by **January 15, 1987,** to Giuseppe Iazeolla, University of Rome II, Dept. of Electronics Engineering, Via O. Raimondo, 00173 Rome, Italy; phone (39)(6) 79792267.

Sixth International Conference on Mathematical Modeling: An Interdisciplinary Integrative Forum on Mathematical Modeling in Engineering, Economics, Biological, Medical, Environmental, Social, and Other Sciences: August 10-13, 1987, St. Louis, Missouri. Papers suitable for 20-minute presentations, papers suitable for 10-minute presentations, and papers for poster sessions are sought. Submit two copies of abstracts (500 words maximum for papers in first category and 200 words maximum for papers in other categories) by **January 15, 1987,** to Ervin Y. Rodin, Dept. of Systems Science and Mathematics, Washington University, Box 1040, St. Louis, MO 63130; (314) 889-5806.

11th Western Educational Computing Conference: November 19-20, 1987, San Francisco, California. Submit two copies of original papers by **April 1, 1987,** to Judah Rosenwald, Extended Education, San Francisco State University, 1600 Holloway, San Francisco, CA 94132.

UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN - SAUDI ARABIA

COLLEGE OF COMPUTER SCIENCE AND ENGINEERING
INFORMATION & COMPUTER SCIENCE DEPARTMENT
COMPUTER ENGINEERING DEPARTMENT

Applications for faculty positions are invited. A Ph.D or an M.S. for the lecturer positions, is required. Evidence of research accomplishment or potential is essential.

Faculty will interact with undergraduate and graduate programs, and will have free access to extensive lab, computer and library facilities.

The University offers attractive salary and benefits which are tax-free.

Send resume with supporting documents to:

UNIVERSITY OF PETROLEUM & MINERALS
HOUSTON OFFICE, DEPARTMENT 509
5718 WESTHEIMER, SUITE 1550
HOUSTON, TEXAS 77057

THE MOVING FRONTIER...

The Art of Prolog

Advanced Programming Techniques

Leon Sterling and Ehud Shapiro

An introduction and reference to Prolog for professionals and an introduction to Logic programming and Prolog programming for students, *The Art of Prolog* is unique in its extremely clear presentation of very advanced topics. Examples show how substantial Prolog programs can be built.
$29.95

ACTORS

A Model of Concurrent Computation in Distributed Systems

Gul Agha

This book addresses cental issues in concurrency, and by producing both a syntactic definition and a denotational model of Hewitt's ACTOR paradigm—a model of computation specifically aimed at constructing and analyzing distributed large-scale parallel systems—it substantially advances the understanding of parallel computation.
$25.00

Knowledge-Based Tutoring

The GUIDON Program

William Clancey

Knowledge-Based Tutoring describes the advantages and difficulties of adapting an expert system for use in teaching and problem solving. This is the first attempt to adapt a rule base for tutoring and opens the door to what will most likely be a dramatic growth in interest in the use of expert systems for teaching.
$30.00

Cellular Automata Machines

A New Environment for Modeling

Tommaso Toffoli and Norman Margolus

Recently, cellular automata machines with the size, speed, and flexibility for general experimentation at a moderate cost have become available to the scientific community. This book discusses their role as natural models for many investigations in physics, combinatorial mathematics, and computer science that deal with systems extended in space and evolving in time according to local laws.
$30.00

Concurrent Prolog

edited by Ehud Shapiro

Concurrent Prolog is a logic programming language designed for concurrent programming and parallel execution. The contributions in this book summarize the development of the language and present techniques, applications, and implementation.
$30.00

Machine Interpretation of Line Drawings

Kokichi Sugihara

"Methods for the intepretation of line drawings in terms of three dimensional objects have been the subject of hot debates in computer graphics and machine vision for two decades. Sugihara presents what is likely to be the last word on the subject: an elegant quantitative method that avoids arbitrary heuristic assumptions."
—Berthold K.P. Horn, MIT
$30.00

Object-Oriented Concurrent Programming

Akinori Yonezawa and Mario Tokoro

This book deals with a major theme of the Japanese Fifth Generation Project, which emphasizes logic programming, parallelism, and distributed systems. It presents a collection of tutorials and research papers on a new programming and design methodology in which the system to be constructed is modeled as a collection of abstract entities called "objects" and concurrent messages passing among objects.
$25.00

Kinematics of Robot Manipulators

edited by J.M. McCarthy

The focus of modern machine design research is the theory and methodology of design of general purpose machines that may be controlled by a computer to perform all the tasks of a set of special-purpose machines. These seventeen contributions chronicle recent activity in the analysis and design of robot manipulators that are the proto-type of these general-purpose machines.
$20.00

Order today—Visa and Mastercard accepted, call (617) 253-2884.

The MIT Press

28 Carleton Street, Cambridge MA 02142

Reader Service Number 5

BOOK REVIEWS

Advanced Database Machine Architecture—David K. Hsiao, ed. (Prentice-Hall, Englewood Cliffs, N. J., 1983, 394 pp., $35)

This book is a result of the second database machine workshop held in San Diego, California, in 1982 and contains 11 papers about nine database machines from five countries, such as VERSO, SABRE, RDBM, IQC (a data stream machine), DBC (a full-text information retrieval system), and MDBS. All of these are "emerging" (i.e., not commercially available, but some are operational) and in the "prototype stage." (At least some of the projects, such as SABRE and MDBS, are alive and well —papers on them were published recently in *Special Interest Group on Management Of Data Record* and *Transactions on Database Systems.* [1-3])

Since database machines are motivated by technological reason (the main motivation: to increase the performance of DBMSs), the approaches taken by the writers to their subject are highly technological, and the architectural ideas presented are diverse and interesting. Quite often the borderline between hardware and software is blurred. The papers are usually well-written and well-presented, and the shorter ones definitely read better.

The papers themselves are like the usual ones describing relational database management systems, and each presents the concepts and architecture of a particular DBMS. Quite a few ideas are interesting in their own right, however, notably, V-relations in VERSO ("universal" relations consisting of "update units"), domain-based internal schema in DBMAC, multiattribute clustering technique in SABRE, prehashing in order to implement joins in DBC, and sort-merge joins in RDBM.

An observation of one of the authors could be used to describe all the papers: "It is the cumulative sum of all the ensuing minor problems of a conventional nature that consume most of the time and money. Since many design specifications have to be frozen a long time before completion, it is not always possible to include new design ideas in the actual implementation." Most of the papers reflect this observation—especially when one takes into account that technological solutions are the focus of all the papers.

The approach taken by all the authors in these papers is a survey rather than a tutorial one. Thus the reader's background should include a knowledge of data structures and database management systems, as well as of some operating systems, networks, and, at times, even elements of hardware design. The longer papers contain a few pages presenting an introduction to the main concepts (especially the one on DBC). However, some other parts of these same papers with their extremely technical details at times read like patents rather than computing science papers.

Overall, the main approach of the book is one of cautious optimism, and as the authors of the paper on IQC remark: "Years of assiduous effort are still needed for the development of database machines that will be widely accepted."

Haim Kilov
Riga, USSR

References

1. P. Valduriez and G. Gardarin, "Join and Semijoin Algorithms for a Multiprocessor Database Machine," *TODS,* Vol. 9, No. 1, March 1984, pp. 133-161.
2. E. Simon and P. Valduriez, "Design and Implementation of an Extensible Integrity Subsystem, *SIGMOD '84 Proceed." SIGMOD Record,* Vol. 14, No. 2, 1984, pp. 9-17.
3. S. Demurjian and D. Hsiao, "Benchmarking Database Systems in Multiple Backend Configurations," *Database Engineering,* Vol. 8, No. 1, March 1985, pp. 29-39.

Programming in micro-PROLOG—Hugh de Saram (Halsted, New York, 1985, 166 pp., $15.95)

The Japanese have selected PROLOG (which is an acronym for Programming in Logic) as the language to be used in the development of the "fifth generation" machine, and the government of Hungary has employed the language to handle tasks varying from the storage and retrieval of structural building codes to software engineering applications. And now, 16 years after the development of the language at the University of Marseilles, it is becoming widely available for the microcomputer market, with versions currently available for CP/M-, MS-DOS-, and UNIX-based systems. de Saram has written a work to introduce a language "theoretically that should make computer programming more available to more people."

The book is small as language textbooks go, and is divided into two major sections entitled Techniques and Applications. The Techniques section is written in keyword-subsection style, but the author omits some keywords such as READ, LNE, and PNT. Where these are omitted, the author refers the reader to the micro-PROLOG reference manual for description and explanation. This section is also peppered with comic-strip style illustrations that seem not to add anything to the understanding of the language. This section, according to the author, "is not an exhaustive account of the system commands of micro-PROLOG."

The Applications section consists of limited amounts of introductory text for games, turtle graphics, database applications, and natural language processing programs written in micro-PROLOG. Unfortunately, though, the results of these programs are left to users to dis-

Recently published books and **new periodicals** may be submitted for review to the book reviews editor:

Francis P. Mathur, Mathematics Department, California State Polytechnic University, 3801 West Temple Avenue, Pomona, CA 91768, (714) 598-4421.

Note: Publications reviewed in this section are not available from the IEEE Computer Society; they must be ordered directly from the publisher.

cover once they translate these programs into the dialect of their own system and successfully run them.

The last major section in the book is an Appendix containing several more programs for the reader to translate.

The text is intended to introduce programmers and hobbyists to micro-PROLOG, and even includes a brief introduction to list processing. Unfortunately, however, programmers may find that file I/O information is limited and hobbyists may find the authors examples difficult to translate into the dialect of their own systems without guidance. Furthermore, the text is designed to be used while seated at a microcomputer keyboard and will not substitute as a generalized introductory text.

I feel that the book would be better titled *A Collection of micro-PROLOG Programs for Spectrum Users.* The book does not offer programming instruction, only programs. If you have a micro-PROLOG reference manual and a system supporting the language, the translation of the programs contained in this text would represent the maximum benefit to be derived.

Vincent L. Boyer, Sr.
E-Systems, Inc., Garland Division

Late Magazines? No Magazines? Membership Status Problems? No Answers To Your Complaints?

Let your Computer Society Ombudsman cut through the red tape for you.

Computer Society
Ombudsman
IEEE Computer Society
10662 Los Vaqueros Circle
Los Alamitos, CA 90720

New Computer Architectures—J. Tiberghien, ed. (Academic Press, London, 1984, 289 pp., $28.50)

This book consists of four parts and was written under the auspices of an International Chair in Computer Science, funded by IBM Belgium in cooperation with the Belgian National Foundation for Scientific Research.

Only Part 1, entitled "Decentralized Computer Architecture" (58 pp., 68 references), delivers what is implied by the book's title. In Part 2, Alvin M. Despain presents his "Notes on Computer Architecture for High Performance" (80 pp., 21 references). Mario R. Barbacci's "Structural and Behavioral Description of Digital Systems" (85 pp., 47 references) constitutes Part 3. These two parts contain tools and techniques related to computer architecture—be it old or new. Finally, Part 4, written by Harold W. Lawson, Jr. and entitled "Computer Architecture Education" (64 pp., 31 references), simply does not belong in this book. The book has been very lightly edited, as evidenced from varying styles of presentation, differing formats for references, and the two-page index.

Professor Treleaven, the author of Part 1, starts by discussing motivations for decentralized computers and types of VLSI systems.[1] He then presents a short overview of new programming languages, divided into functional, actor, and logic classes. The remaining 70 percent of this part (program and machine organization) comes mostly from the author's previous survey paper.[2] The novice will find this discussion too cursory to be of much help, while the expert would certainly prefer the more comprehensive survey papers.

Professor Despain's contribution reads like the detailed outline for a forthcoming book, with introductory material on defining computer architecture, the role of computer architects, architectural principles (including heuristic search of the design space and the Turing Machine), and history of computer technology given a 25 percent share. This is followed by discussions on models of computation (18 pp.), serial-parallel program transformations (14 pp.), recurrences (taken from the works of Kuck[3]) (9 pp.), and parallel prefix calculations (applicable to a restricted class of nonlinear recurrences) (17 pp.).

Professor Barbacci presents an overview of computer hardware description languages, or CHDL's, in the first fifth of Part 3 and then devotes some 50 pages to ISP-level (for instruction set processor) description and evaluation of computer systems.[4] What spoils this best organized and most readable part of the book is a concluding 12-page description of the SPICE personal computing project at Carnegie-Mellon University—a description totally unrelated to ISP's or CHDL's.

Professor Lawson's contribution should have been entitled "An Approach to Teaching Computer Fundamentals." It reads very much like promotional literature for his two books; one published[5] and the other forthcoming. His ideas, though probably original, do not belong in this book and could have been presented in less than half the space. For example, one hardly needs a four-page linguistic argument to become convinced that graphical architecture and algorithm descriptions are easier to comprehend. A partial list of the sections in this part of the book should convey its nonrelevance: The Human as an Information Processor (11 pp.), The Role of Language and Representation (4 pp.), and The Creative Processes of Individuals and Groups (11 pp.).

Overall, I cannot recommend this book to anybody. Whatever valuable material there is to it can be found elsewhere, especially in the other works of the authors. This book neither has coherence nor gives the reader deeper insight in its treatment of the loosely related topics covered.

Behrooz Parhami
University of Waterloo

References

1. P. C. Treleaven, "VLSI Processor Architectures," *Computer,* Vol. 15, No. 6, June 1982, pp. 33-45.
2. P. C. Treleaven, D. R. Brownbridge, and R. P. Hopkins, "Data-Driven and Demand-Driven Computer Architecture," *Computing Surveys,* Vol. 14, No. 1, Mar. 1982, pp. 93-143.
3. D. J. Kuck, *The Structure of Computers and Computations,* Wiley, New York, 1978.
4. M. R. Barbacci and D. P. Siewiorek, *The Design and Analysis of Instruction Set Processors,* McGraw-Hill, New York, 1982.
5. H. W. Lawson, Jr., *Understanding Computer Systems,* Computer Science Press, Rockville, Maryland, 1982, and Lawson Publishing Co., Linkoping, Sweden, 1979.

Organization and Management of a Computer Center—Martin Graef and Reinald Greiller (Macmillan Publ. Co., New York, 1985, 384 pp., $34.95)

Martin Graef and Reinald Greiller's book contains a detailed description of the activities involved in a computer center, from it's inception to the point of maturity. The volume is organized into 10 chapters covering the widest variety of computer center concerns. The subjects and the sequence of their presentation are: Types of Computer Centers and Their Tasks; Organization; Operations Organization; Data Protection and Data Security; Computer Center Personnel; Aspects of Communication; Planning Control; Planning and Implementing a Computer Center; Cost and Price Planning; and Computer Center Handbook.

Each of the chapters is quite detailed and usually will contain discussions of the advantages and disadvantages relating to a particular topic. Many of the chapters provide the reader with methods to quantify areas of concern. For example, Chapter 7, Planning and Control, has formulas for the efficiency rate of the central processing unit, the periphery efficiency rate, the efficiency rate of the main storage unit, and others.

Throughout the volume there are many helpful checklists, sample forms, lists of considerations in planning, and sample position descriptions. For example, in Chapter 8, Planning and Implementing a Computer Center, there is a discussion of factors to consider in space planning. This discussion is divided into factors relating to the computer room, data entry room, shipping room, stockroom, and archive for magnetic storage devices. These sections include proposed floor plans, square footage recommendations, environmental conditioning recommendations, and so forth.

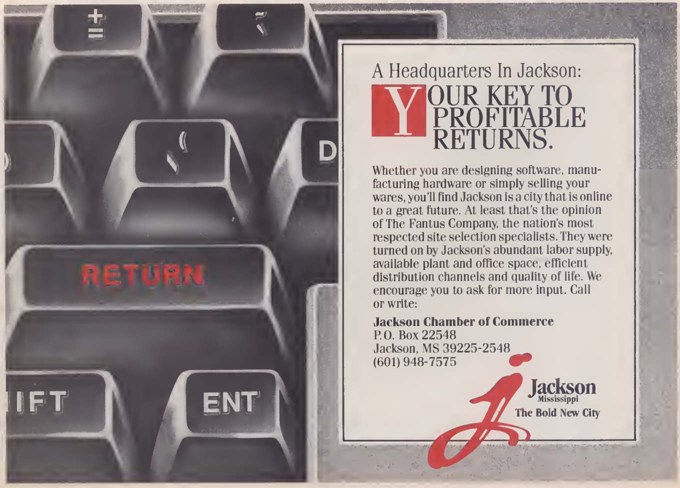
There are two areas where the book appears a little weak. The first is the language. The book is a translation from the German so it is difficult to determine the source of the weakness, but an example is on page 166, where the authors are describing causes for loss of data and programs and state: "This includes losses incurred through deliberate theft or the intervention of a higher power—for example, fire, water, or radioactive damage." Another example is on page 3, where the authors state: "Two points of view are possible with respect to the computer center's disposition towards job orders." The sentences are often long, and because of the way the language is used, the volume is difficult to read. The other area of weakness is the limited consideration of distributed processing. The basic orientation of the volume is toward the central computer room, complete with data entry clerks working for the data processing unit.

In summary, this volume is very complete in the discussion of all aspects of traditional computer center planning and operation. It should be of assistance to those in the process of planning or operating a large computer center, enabling them to plan without leaving "any stone unturned," or operating with the greatest efficiency.

Steven J. Altig
Internal Revenue Service

Erratum

In the July 1986 issue of *Computer*, the title of a book reviewed by Paul Gray appeared as *Review of Micro Database Management: Practical Techniques for Application Development.* It should have been *Micro Database Management: Practical Techniques for Application Development.*



A Headquarters In Jackson:

YOUR KEY TO PROFITABLE RETURNS.

Whether you are designing software, manufacturing hardware or simply selling your wares, you'll find Jackson is a city that is online to a great future. At least that's the opinion of The Fantus Company, the nation's most respected site selection specialists. They were turned on by Jackson's abundant labor supply, available plant and office space, efficient distribution channels and quality of life. We encourage you to ask for more input. Call or write:

Jackson Chamber of Commerce
P.O. Box 22548
Jackson, MS 39225-2548
(601) 948-7575

Jackson
Mississippi
The Bold New City

Reader Service Number 6

NEW PRODUCTS

A sampling of the GaAs market

Nancy Hays, Assistant Editor

Around thirty American companies are currently involved in the production of GaAs integrated circuits. Most do not offer products for sale on the open market, although the number is growing. Instead, many companies have developed customized GaAs ICs for use in their own products. Wayne Moyers of RCA Advanced Technology Laboratories calls the industry immature, pushed by the military. (For a discussion of the military's role, see the article in this issue, "DARPA, SDI, and GaAs," by Karp and Roosild.) Certainly the expense of design and implementation relegates many GaAs products to the research and development labs. (For a discussion of the characteristics of GaAs technology, see the guest editor's introduction.)

A handful of companies currently offer GaAs products for sale, notably Anadigics and Vitesse Electronics, with Gain Electronics scheduled to offer products in 1987. Also notable is Pacific Monolithics, which reputedly provides GaAs systems from the top down.

Foundries producing GaAs products in most cases started in-house to service R&D demands, or commercially to produce custom and semicustom chips for other companies. Foundry services vary, but many now include tutorials to aid in the design of GaAs ICs. Each foundry has its own rules for designers to follow, since masks supplied by the customer require adjustment of the production process. Many foundries also handle testing and packaging of the chips.

Notable are the foundries that supply standard as well as custom products. GigaBit Logic offers the PicoLogic product family, which the company calls building blocks for the system designer, and plans to offer standard cells shortly. Harris Microwave Semiconductor, one of the earliest GaAs foundries, claims that standard products are the company's largest business, surpassing their custom foundry service. The first commercial foundry service (since 1984), TriQuint Semiconductor, offers a medium-scale integration configurable cell array called Q-Chip for development and standard GaAs IC components for digital and microwave applications. Vitesse Electronics offers standard logic and memory large-scale integration ICs.

Companies that produce GaAs products for the military generally have not become involved in the commercial market, focusing instead on research and development and the military market. Such might include ITT, RCA, Texas Instruments, and McDonnell-Douglas (which has deliberately tried to manufacture radiation-hardened GaAs products). Others, such as Honeywell, are moving to expand and commercialize their production capabilities.

The accompanying chart lists some of the companies involved in GaAs production, whether commercial or private. It is by no means inclusive. The column

COMPANY	CONTACT	FOCUS	RS NO.
Adams-Russell Electronics 80 Cambridge St. Burlington, MA 01803	Michael Smith (617) 273-3333	Commercial production of MMICs, foundry services, and component manufacture. Offers multiproject use of wafers.	40
Anadigics 35 Technology Dr. Warren, NJ 07060	Michael Gagnon (201) 668-5000	Commercial production of microwave, analog, and digital preprocessing GaAs ICs, including standard and custom products. A systems approach with integrated chip sets.	41
Ford Microelectronics 10340 State Highway 83 N. Colorado Springs, CO 80908	Bob Gisburne (303) 528-7624	Commercial manufacture of MSI and LSI digital and analog GaAs products, including standard products, foundry services, custom products, and gate arrays.	42
Gain Electronics 22 Chubb Way Summerville, NJ 08876	Mike Logan (201) 526-7111	Commercial production of selectively doped heterostructure transistors, custom and semicustom circuits.	43
GE Electronics Laboratory PO Box 4840-EP3 Syracuse, NY 13221	Bob Fink (315) 456-3922	R&D and prototyping of analog GaAs MMICs for DOD applications, with gates ranging from 0.25 to 1 micron. A captive source focusing on low noise devices. Also digital, CAD tools, cell libraries, and custom circuits for GE applications.	44
GigaBit Logic 1908 Oak Terrace Lane Newbury Park, CA 91320	Stewart Sando (805) 499-0610	Commercial production of standard components and semicustom design to permit building complete system solutions. Provides GaAs building blocks for designers. Foundry services for custom products.	45
Harris Microwave Semiconductor 1530 McCarthy Blvd. Milpitas, CA 95035	Bruce Hoffman (408) 262-2222	An early commercial producer stressing vertical integration and experience in standard and custom products. Foundry services for custom circuits. FETs, digital and analog ICs, FET amps.	46
Honeywell 830 E. Arapaho Rd. Richardson, TX 75081	Bill Woodruff (214) 234-4271	R&D and production of digital, semicustom, and LSI GaAs products. Some standard products. Serves DOD and handles a DARPA GaAs LSI pilot line. Offers a multiproject test chip option.	47
ITT GaAs Technology Ctr. 7625 Plantation Rd. Roanoke, VA 24019	Edward Griffin (703) 563-8600	Production of monolithic GaAs ICs for military system insertion. A captive foundry, but increasing non-ITT customers. Development and implementation, mainly for military applications.	48

labeled "Focus" summarizes each company's perception of its major involvement in GaAs technology. Readers interested in learning more about a specific company's services can contact the company representative listed or circle the appropriate reader service number on the reader service card at the back of the magazine.

GaAs MMIC library

EEsof and TriQuint Semiconductor have announced the Touchstone/TriQuint MMIC element library, which incorporates TriQuint's GaAs custom MMIC foundry models into EEsof's Touchstone software for the design and analysis of linear microwave circuits.

The library covers common MMIC components, including MESFETs, diodes, inductors, capacitors, and resistors. Combined with Touchstone, the library allows users to specify MMIC models in terms of TriQuint's manufacturing database processing parameters.

For more information, contact EEsof, Inc., 31194 La Baya Dr., Westlake Village, CA 91362; (818) 991-7530.

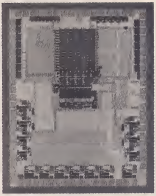
Reader Service Number 60

Vitesse offers LSI GaAs chips

Vitesse Electronics has announced a family of digital LSI GaAs components, including the VE29G01, a 4-bit microprocessor; the VE29G02, a lookahead carry generator; and the VE29G10A, a 12-bit microcontroller. The family is microcode compatible with Advanced Micro Devices' Am2900 series, and compatible with industry standard 100K ECL I/O. The products are made using Vitesse's proprietary second generation E/D-mode technology.

The VE29G01 consists of a 16-word by 4-bit two-port RAM, an eight-function ALU, and associated shifting, decoding, and multiplexing circuitry. A 4-bit add reputedly takes 15 ns and a read-modify-write takes 14 ns. Maximum clock speed measures 72 MHz. The device comes in a 52-pin ceramic leadless chip carrier. Engineering samples cost $225. Production parts will be available in 1987.

The VE29G02 accepts up to four pairs of carry propagate and generate signals plus a carry input signal. It can provide carry lookahead across four groups of binary ALUs. A carry propagate cycle reputedly takes 5 ns. The device comes in a 28-pin ceramic leadless chip carrier. Samples will be available in the first quarter of 1987 for $435.



VE29G01

The VE29G10A supports sequential access and permits conditional branching to any microinstruction within its 4096-word range. A LIFO stack provides subroutine return linkage and looping. Address generation reputedly takes 10 ns, while cycle time supported is 25 ns. Maximum clock speed measures 100 MHz. The device comes in a 52-pin ceramic leadless chip carrier. Samples will be available in the first quarter of 1987 for $475.

For more information, contact Vitesse Electronics Corp., 741 Calle Plano, Camarillo, CA 93010; (805) 388-3700.

Reader Service Number 61

COMPANY	CONTACT	FOCUS	RS NO.
McDonnell-Douglas Microelectronics PO Box 516 St. Louis, MO 63166	Owen L. Peters (314) 232-4961	Digital GaAs technology for military and aerospace applications. Specializes in radiation-hard circuits, memories, gate arrays, microprocessors, etc. No commercial contracts.	49
Microwave Associates (M/A-COM) 100 Chelmsford St. Lowell, MA 01851	Joanne Dalton (617) 937-2800	Largest US producer of GaAs wafers. R&D and manufacture of ICs, especially MMICs, but also digital ICs to enhance analog functions. Offers foundry services and discrete FETs.	50
Microwave Semiconductor 100 School House Rd. Somerset, NJ 08873	Jim Dilorenzo (201) 563-6210	R&D of discrete microwave, MMIC, and digital ICs. Custom products for telecommunications and military applications. Standard products are discrete. In-house production of GaAs wafers. Planned production of ICs for fiber optics (1987).	51
Pacific Monolithics 245 Santa Ana Court Sunnyvale, CA 94086	Doug Lockie (408) 732-8000	Commercial and military production of microwave signal processing subsystems. ASIC supplier of MMICs, design and packaging of analog microwave products. Mostly custom products, but offers about 50 standard cells and 8 standard products.	52
RCA Advanced Technology Labs Mail Station 145-3 Moorestown, NJ 08057	Wayne Moyers (609) 866-6427	R&D house, custom design for military applications. Handles both digital and analog GaAs ICs. Planning an 8-bit CPU for 1987, based on RISC architecture with a net throughput of 100 MIPS.	53
TI Defense Systems & Electronics PO Box 660246 Dallas, TX 75266	Philip Congdon (214) 480-1944	MMICs and digital GaAs ICs for military applications. An in-house foundry plans to offer commercial services. Planning a 32-bit CPU using RISC technology.	54
TriQuint Semiconductor Group 700, PO Box 4935 Beaverton, OR 97076	Dennis Powell (503) 629-4227	The first full-service GaAs foundry. Claims the largest number of GaAs products and services offered commercially. Provides custom and semicustom analog and digital chips, design tools and training. Standard components for system integration, plus ASICs to specification.	55
Vitesse Electronics 741 Calle Plano Camarillo, CA 93010	Jim Brye (805) 388-3700	Commercial production of LSI-level GaAs parts (IC Division). Standard logic and memory products available. Foundry services include design aid, packaging, and testing. Building minisupercomputers using GaAs chips as available.	56

New from Data General

Data General Corp. has announced third-generation software environments for its TEO software family, a workstation compatible with the IBM PC-AT, and a midrange superminicomputer.

TEO/Electronics software is a third-generation electronics design automation software system that reputedly integrates advanced engineering tools with office functions. It is compatible with Data General's CEO office automation software. TEO/3D software is a third-generation computer-aided design and modeling program that also integrates CEO-compatible office automation functions. Both are designed for integration with Data General's DS/7000 family of engineering workstations.

The Dasher/286 workstation is software compatible with the IBM PC-AT. It features an Intel 80286 processor, standard memory of 640K bytes of RAM, and CEO office automation software for the MS-DOS environment. The workstation operates at 10 MHz, but can switch to 6 MHz when necessary. Standard features include one 5.25-inch disk drive with 1.2M bytes of memory, two RS-232/422 serial asynchronous ports, one Centronics-compatible parallel printer port, and six standard expansion slots. The Dasher/286 comes with MS-DOS 3.1. The base system costs $3395, $4095 with a 20M-byte hard disk.

The Eclipse MV/7800 computer has a 32-bit architecture based on a six-member chip set consisting of four proprietary VLSI NMOS circuits, a CMOS gate array, and an ECL gate array. An MV/7800 C rackmount model with a 2M-byte system board costs $19,050, with a monthly maintenance charge of $90. The MV/7800 U upgrades range in price from $10,000 to $17,500.

For more information, contact Data General, 4400 Computer Dr., Westboro, MA 01580; (617) 366-8911.

Reader Service Number 37

Hi-res controllers go up to 2048 by 2048 pixels

Metheus has announced two graphics display controllers reputedly capable of operating at up to 2048 by 2048 pixels, 60 Hz noninterlaced with an RGB color monitor. The Omega 3700 controllers belong to the Omega 3000 Series. The Omega 3710 and Omega 3720 offer up to 256 simultaneous colors from a palette of 262,144. A double-buffering option handles real-time or dynamic applications.

The base price of the Omega 3710 is $29,950. Typical Omega 3720 systems cost between $35,000 and $45,000. For more information, contact Metheus Corp., 5510 N.E. Elam Young Pkwy., Hillsboro, OR 97124; (503) 640-8000.

Reader Service Number 38

Epics 20/20 handles batch and interactive composition

Varityper has announced a workstation for the Epics Composition System. The Epics 20/20 expands the range of applications of the Epics-XL Composition System. In the interactive mode, it provides a character-by-character true type display of work in progress.

The 17-inch diagonal screen is divided into two windows. Typesetting parameters and the text appear in the text-entry window. The job elements, shown in exact size and position, appear in the display window in the actual typestyle used in final output. The size and display characteristics of each window can be modified according to operator preference.

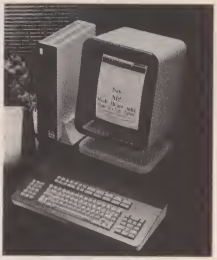
Three prompt and status lines keep operators apprised of relevant system functions. Type gauges on the monitor help the operator measure his or her work, whether in metric, English, or picas and points.

Other interactive capabilities include slewing, to automatically increase or decrease point size, line length, leading, and indents. The screen display can be enlarged or reduced, and zooming covers the range of 25 percent to 400 percent. A panning feature covers work that exceeds the dimensions of the display.

All composition features are compatible between Epics-XL terminals and the 20/20 workstation. The 20/20 is available as an option for new systems or as a field upgrade to the Epics-XL. No price given.

For more information, contact Varityper, 11 Mt. Pleasant Ave., E. Hanover, NJ, 07936; (201) 887-8000.

Reader Service Number 39



The **Epics 20/20** workstation from **Varityper**.

Gold Hill offers expert system for PC ATs

Gold Hill Computers has announced an expert system building environment called Acorn, based on the Golden Common Lisp 286 Developer. A software product for the IBM PC AT, Acorn provides a flexible paradigm for building expert systems.

The program is organized in multiple levels. The top level, an expert system shell, provides the developer with an interface for describing objects and expressing rules and relationships. The middle level, the toolkit, consists of the knowledge representation language. The lower level is Common Lisp.

Developers can build applications using up to 15M bytes of memory, according to the company. An on-line tutorial introduces expert system development plus example applications. Moreover, users can access data stored in dBase II and III and Lotus 1-2-3 directly, or integrate with programs written in conventional programming languages, such as C. The help system reputedly matches help messages to the user's level of sophistication and to the context.

Product components include a complete Lisp environment with interpreter, compiler, and GMACS editor; inference engine, knowledge representation language, graphics module, interface building facilities, and browser; expert system tutorial; on-line help system; and documentation.

Technical features include integrated forward and backward chaining, frame-based knowledge representation, power screen generation, mouse support, hooks to PC tools, certainty factors, and explanation facility.

Acorn will be available in the first quarter of 1987 for $5000. Contact Gold Hill Computers, 163 Harvard St., Cambridge, MA 02139; (617) 492-2071.

Reader Service Number 62

FPS offers minisupers

Floating Point Systems has announced the independent Superstation and Superserver minisupercomputers and the entry-level M64/10 front-end compatible minisupercomputer.

The M64/320 Superstation features surface-mount boards and a reputed peak performance of 6 Mflops. The M64/330 Superstation features a reputed peak performance of 12 Mflops. Prices range from $187,000 to $236,000.

The M64/220 Superserver provides a reputed peak performance of 6 Mflops and the M64/230, 12 Mflops. Prices range from $148,000 to $236,000.

The FPS M64 Series of minisupercomputers includes other models delivering from 11 to 341 Mflops performance at prices from $137,000 to more than $1 million.

For more information, contact Floating Point Systems, Box 23489, Portland, OR 97223; (503) 641-3151.

Reader Service Number 63

Call for Papers

Second International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS–II)

Rickey's Hyatt House
Palo Alto, California, October 5–8, 1987

SIGARCH
SIGPLAN
SIGOPS

Sponsored by ACM and the Computer Society of the IEEE

TC MM
TC VLSI
TC OS

This is a reprise of the highly successful symposium held in 1982. Many of the ideas discussed at that time are now reaching the marketplace. This time, we intend to focus on experimental analyses of architectures and performance evaluations of prototype systems, including application–specific designs. The time is ripe to review progress and identify future directions.

Full papers (approximately 5000 words) are solicited in:

- Language and OS–Influenced Processor Designs
- Architecture–Influenced Compiler and OS Design
- Experimental Analysis of Instruction Set and Other Architectural Design Choices
- Performance Evaluation of Prototype and Newly Available Commercial Systems

We also solicit short (1000 word) high quality technical abstracts on innovative commercial and prototype systems that address topics of relevance to this conference. Full presentations will be expected.

Five copies of submitted papers should be sent to the program chairman at the address below, and will be accepted for evaluation until **February 1, 1987.** Submissions will be read by members and designated reviewers of the program committee:

Randy Katz *U. C. Berkeley (chairman)*
Forest Baskett *Silicon Graphics*
David Ditzel *AT&T Bell Laboratories*
James Goodman *University of Wisconsin*
John Hennessy *Stanford University*
Edward McCreight *Xerox PARC*
Steven Muchnick *Sun Microsystems*
Richard Sites *DEC Hudson*
Alan Smith *U. C. Berkeley*
Chuck Thacker *DEC Systems Res. Center*
Philip Treleaven *University College London*
Mario Tokoro *Keio University*

Authors will be notified of acceptance or rejection by **May 15, 1987.** Accepted papers must be typed on special forms and received by the program chairman by **July 1, 1987.**

A number of special activities are planned in conjunction with the conference, including tutorials, invited papers, and a special journal issue. The best submitted papers will be considered for publication in this issue.

Program Chairman
Randy H. Katz
Computer Sciences Division
U. C. Berkeley, Evans Hall
Berkeley, CA 94720

General Chairman
Martin Freeman
Signetics Corp.
811 E. Arques Ave.
Sunnyvale, CA 94086

Tandem adds NonStops

Tandem has added two models to its NonStop computer systems, the EXT10 and the EXT25. The base NonStop EXT10 system costs $82,500. It is an expandable fault-tolerant transaction processing system that handles up to 4.3 transactions per second. The midrange NonStop EXT25 costs $325,000. It reputedly processes up to 11 transactions per second. Expanded systems of the two models process up to 8.6 and 22 transactions per second respectively.

The basic systems include two processors; 8M bytes of DRAM for the EXT10 and 16M bytes for the EXT25; 128M-byte Winchester 8-inch drives; disk controllers; cartridge magnetic tape drives; the 6105 communication controller; power supplies; and an operations and service processor.

For more information, contact Tandem Computers Inc., 19191 Vallco Pkwy., Location 4-40, Cupertino, CA 95014-2599; (408) 725-6000.

Reader Service Number 64

Apollo has Turbo workstations

Apollo Computer Inc. has announced two turbo color graphics workstations, the DN570 Turbo for two-dimensional graphics and the DN580 Turbo for real-time three-dimensional graphics.

The workstations feature a 32-bit VLSI CPU based on a MC68020 processor and MC68881 floating-point coprocessor, up to 16M bytes of error checking and correction memory, support for up to 2G bytes of virtual address space per user process, and a floating-point accelerator option called FPX ($5900).

Prices range from $43,900 to $66,900. Contact Apollo Computer Inc., 330 Billerica Rd., Chelmsford, MA 01824; (617) 256-6600.

Reader Service Number 65

dBase now available on the Macintosh

Ashton-Tate has announced dBase Mac for the Apple Macintosh. The company claims that this relational database management system combines the power of Ashton-Tate's dBase products with the ease of learning and presentation available on the Macintosh. The system costs $495.

Features include pop-up menus and dialog boxes, plus templates for data entry screens and reports. Users can link up to 36 data files through common fields. The program can access IBM-compatible dBase data files through ASCII text files. It supports a variety of field and data types. The file and record capacity is limited by the size of the operating system.

An advanced procedural language permits turnkey applications. The protect mode preserves applications from user alteration.

dBase Mac requires a Macintosh with 512K bytes of RAM, or a Macintosh Plus. It requires a minimum of one 800K-bit floppy disk drive and may be installed on a number of hard disks.

Contact Ashton-Tate, 20101 Hamilton Ave., Torrance, CA 90502-1319; (213) 329-8000.

Reader Service Number 66

Cray adds X-MP models

Cray Research has added two models to the Cray X-MP line. The Cray X-MP/44 has four CPUs sharing a four-million-word ECL bipolar memory arranged in 32 interleaved banks. It costs $12 million. The Cray X-MP/22 has two CPUs sharing two million words of MOS memory arranged in 16 banks. It costs $7 million. Contact Cray Research, Inc., 608 Second Ave. S., Minneapolis, MN 55402; (612) 333-5889.

Reader Service Number 67

VAX models use VAXBI

Digital Equipment Corp. has added the VAX 8550 and the VAX 8700 to its second-generation computers using the VAXBI architecture. The systems come in seven configurations, with base prices ranging from $364,000 for the VAX 8550 and from $433,000 for the VAX 8700. Contact DEC, Maynard, MA 01754-2571.

Reader Service Number 68

Turbo Pascal for the Mac

Borland International has introduced Turbo Pascal for the Apple Macintosh personal computer. Key features include implementation of pull-down menus and the Macintosh point-and-click interface; the ability to run large programs using a unit structure based on modules called "units" created by separate compilation of routines; multiple editing windows; reputed compilation speeds of over 12,000 lines per minute; compatibility with Lisa Pascal; compatibility with the hierarchical file system of the Macintosh; search and change features in the editor; and compilation options, including the choice to compile to disk or memory, or to compile and run.

Turbo Pascal for the Mac is not copy protected. It requires at least 256K bits of memory, along with a 400K-bit disk drive. It will be available November 15, 1986 for an introductory price of $99.95 (through March 15, 1987).

For more information, contact Borland International, Inc., 4585 Scotts Valley Dr., Scotts Valley, CA 95066; (408) 438-8400.

Reader Service Number 69

NEW LITERATURE

Three-volume CAD/CAM directory. Volume I, *International Computer-Aided Manufacturing (CAM) Directory,* has specification-oriented descriptions of hardware for CAM applications; II, *International Computer-Aided Design (CAD) Directory,* of CAD hardware; and III, *International CAD/CAM Software Directory,* of CAD/CAM software. Technical Database Corp., PO Box 720, Conroe, TX 77305; (713) 439-1687; Vols. I and II, $39 each; Vol. III, $34.

Report on portable computers. "Portable Computers" gives the market outlook for all categories of portables and covers such topics as distribution channels, application needs, and market forecasts. Frost & Sullivan, Inc., 106 Fulton St., New York, NY 10038-2786; (212) 233-1080; $1875.

Nontraditional Machining. Proceedings of a 1985 conference in part sponsored by the American Society for Metals. This 160-page book focuses on applications of the latest technology in the industrial workplace, including laser beam, electrical discharge, and ultrasonically assisted machining. Communications, ASM, Metals Park, OH 44073; (216) 338-5151; $62 (ASM members, $49.60).

The IBM Token-Ring Product Guide. Distilled compendium of IBM's Token-Ring product announcements organized into three sections of Wiring, Token-Ring Connectivity, and Token-Ring Software. Product listings are indexed and cross-referenced. Architecture Technology Corp., PO Box 24344, Minneapolis, MN 55424; (612) 935-2035; $125.

Guide to computer law. Covers management and legal issues of acquiring, marketing, and managing computer hardware, software, and telecommunications. The two-volume *Bernacchi on Computer Law* has a checklist of contract terms and a sample project timetable. Little, Brown & Co., 34 Beacon St., Boston, MA 02106; $160.

A Guide to Alternative Careers. David E. Dougherty outlines techniques for transforming technical expertise into substantial financial gain. Such topics as getting patent protection, writing claims, and evaluating the profitability of ideas are covered. John Wiley & Sons, Inc., 605 Third Ave., New York, NY 10158; (212) 850-6000; $22.95.

computer networking symposium

Loews L'enfant Plaza
Washington, DC
November 17-18, 1986

Conference Chairperson:
Tuncay Saydam, University of Delaware
Technical Program Chairperson:
George Chang, Bell Communications Research

A Symposium Sponsored by:

The Computer Society of the IEEE

THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, INC.

TUTORIALS

MONDAY, NOVEMBER 17, 1986

LOCAL NETWORK TECHNOLOGY, William Stallings, Ph.D., Management Consultant

This tutorial is a survey of the state of the art of local networks. The key technology areas are explored, including: transmission media, topologies, protocols, switching techniques, hardware/software interface, and performance. Recent developments in optical fiber technology for local networks are discussed, including a look at the emerging Fiber Distributed Data Interface (FDDI) standard. Related design issues such as local network servers, network control, reliability, and security are also discussed.

OSI AND ISDN PROTOCOLS AND SERVICES, Bryan Whittle, Bell Communications Research and Wayne Davison, Research Libraries Group, Inc.

The seminar will examine (1) OSI objective, ISDN objectives and relations between those two sets of objectives; (2) OSI and ISDN functional and protocol architecture and relations between them; (3) OSI service and ISDN services and relations between them–the viewpoints of "protocol layer services" and "public network to end-user services" will be related; (4) application scenarios which illustrate how an end-user can apply OSI and ISDN to serve data applications and to serve applications that require simultaneous communication of data and voice; (5) current OSI and ISDN standardization and implementation activities, trends in those activities and opportunities for increased synergy between those activities.

TECHNICAL PROGRAM

TUESDAY, NOVEMBER 18, 1986

SESSION A1: Protocol, Robert Rosenthal, NBS

SESSION A2: Protocol II, Emile Soueid, Bell Communications Research

SESSION A3: Routing and Flow Control, Guy Omidyar, Riverside Research Institute

SESSION B1: LAN/MAN, Robert Klessig, Bell Communications Research

SESSION B2: Packet Switched Networks, John W. Mark, University of Waterloo

SESSION B3: Internetworking, Susan Poh, IBM

SESSION C1. Performance Analysis & Modeling, Adarshpal Sethi, University of Delaware

SESSION C2: Distributed Systems/Algorithms, Ravi Mazumda, Columbia University

SESSION C3: Measurments, Kathy Huber, BBN

COMPUTER NETWORKING SYMPOSIUM '86

Loews L'enfant Plaza, Washington, D C
ADVANCE REGISTRATION FORM
November 17-18, 1986

Complete and return this form with your check made payable to Computer Networking '86 to:

Computer Networking '86
1730 Massachusetts Ave., N.W.
Washington, D.C. 20036-1903
Telephone: (202) 371-0101
TWX 7108250437 IEEECOMPSO

Please check appropriate box:	**Advance Registration (prior to 10/31/86)**		**Late Registration (after 10/31/86)**	
	Member	Non-member	**Member**	Non-member
Tutorial 1 Local Network Technology	☐ $125	☐ $160	☐ $150	☐ $190
Tutorial 2 OSI and ISDN Protocols and Services	☐ $125	☐ $160	☐ $150	☐ $190
Conference	☐ $125	☐ $155	☐ $150	☐ $190
Conference & Tutorial 1	☐ $250	☐ $315	☐ $300	☐ $380
Conference & Tutorial 2	☐ $250	☐ $315	☐ $300	☐ $380

NOTE: Student discount available for conference only (Fee $50)
Student Registration does not include proceedings.
To receive student discount students must not be employed full-time

- Requests for refunds must be received in writing no later than 10/31/86
- Tutorial Registration includes luncheon coffee breads and tutorial notes
- Conference registration includes one copy of the proceedings and luncheon
- **People who register at the door for the tutorial will not be guaranteed a copy of the tutorial notes.**

IEEE or Computer Society Number

☐ Check Enclosed ☐ VISA ☐ MasterCard ☐ American Express

Card Number Exp Date

Signature

Name

Title

Company Name

Street Address

City/State/Zip

Lowes L'enfant Plaza

Hotel Reservation Form

Complete & Mail this form to
Computer Networking '87
Loews L'enfant Plaza
480 L'enfant Plaza S.W.
Washington, D.C. 20024
(202) 484-1000

Please circle accommodations required

Single	$100 per nite
Double	$120 per nite

Loews L enfant Plaza is centrally located in the Nation s Capital a short walk from Capitol Hill the Smithsonian Institute the Air and Space Museum the National Gallery of Art and the Hirshhorn Museum The largest Metro subway stop in Washington D C is located directly beneath the hotel and connects you with theater shopping business and government centers We have the perfect location for touring or business

All rates are subject to 10% city taxes and $1.00 per day per room occupancy tax.

Arrival Date A M P M

Departure Date A M P M

ALL MAJOR CREDIT CARDS ACCEPTED

☐ VISA ☐ MasterCard ☐ American Express ☐ Other

Card Number

Exp Date

Signature

Name

Company

Address

City/State/Zip

Please indicate whether you will be a late arrival & wish to guarantee your room. If so; a one night's deposit will be required for the guarantee and will be charged to your credit card. The room block will only be held until October 27. If you make reservations after that date, it will strictly be on a space available basis.

Recent Microsystem Announcements

MODEL AND FUNCTION	COMPANY	COMMENTS	R.S. NO.
Modem 9600	Black Box Corp. PO Box 12800 Pittsburgh, PA 15241 (412) 746-5530	Has manually selectable data rates of 4300, 7200, or 9600 bps, works with IBM PC XT, AT, and compatibles, and interfaces with minis or mainframes through its RS-232-C connection to an async port or protocol converter. Built-in CRC error detection/correction. Cost: $995.	70
BDH-1 BUBBL-DEK, BPH-1 BUBBL-PAC Bubble-memory system and cassettes	Bubbl-tec 6805 Sierra Ct. Dublin, CA 94568 (415) 829-8700	Solid-state and nonvolatile, this one- or two-cartridge magnetic-bubble memory system housed in glass-filled Lexan frame fits the same mounting holes as a standard full-height 5.25-inch floppy disk drive. Features an access time of 14 ms and data transfer rate of 90K bits. Controller is optional. Each front-loadable BPH-1 BUBBL-PAC cartridge provides 128K bytes of storage. Cost: $1324 for the BUBBL-DEK, $175 for BUBBL-PAC.	71
VME ISIO-1 Serial I/O board	Force Computers, Inc. 727 University Ave. Los Gatos, CA 95030 (408) 354-3410	Offers eight multiprotocol serial channels that handle 9.6K baud with programmable rates from 110 to 38.4K baud. Features 128K-byte dual-ported RAM, sockets for up to 128K-byte EPROM, and on-board 10-MHz 68010 processor. Cost: $2195.	72
HK68/V2F Microcomputer	Heurikon Corp. 3201 Latham Dr. Madison, WI 53713 (608) 271-8700	This VME microcomputer offers a 25-MHz Motorola 68020 microprocessor unit and comes with 4M bytes of DRAM, 128K bytes of EPROM and SRAM, and an RS-232 serial port. Cost: $1195 (100's).	73
5842 Modem	IBM 1133 Westchester Ave. White Plains, NY 10604	This microprocessor-based stand-alone modem supports asynchronous or synchronous data transmission up to 2400 bps across CCITT V.24/V.28 (EIA RS-232-C/RS-334) interfaces. With built-in diagnostics, automatic or manual dial capabilities, and storage for up to 20 telephone directory entries. Cost: $719.	74
286 Express Card Accelerator card	May-Craft Information Systems, Inc. 4312 Beltwood Pkwy. South Dallas, TX 75244 (214) 392-3766	Half-slot accelerator card for the IBM PC and PC XT with an 80286 microprocessor and 8K-byte cache memory. Has connection for an 80287 floating-point coprocessor. Cost: $595.	75
MasterForth Forth environment	Micromotion 8726 S. Sepulveda Blvd., #A171 Los Angeles, CA 90045 (213) 821-4340	Consisting of macroassembler, a full-file interface, string package, screen editor, and debugger, this implementation of the Forth programming language (Forth-83) can be used in the Macintosh, the IBM PC family, the Apple II series, the Commodore 64, and the Z80 CP/M. Includes *Mastering Forth,* a book on the Forth-83 dialect. Cost: $100-$125.	76
NS865-16 Memory board	National Semiconductor Corp. 2900 Semiconductor Dr. PO Box 58090 Santa Clara, CA 95052-8090 (408) 733-2600	A 16M-byte memory card compatible with DEC's VAX 8600/8650 computers. Features on-board diagnostics and spare DRAM and is equipped with an on-line/off-line switch with an LED indicator. Price not listed.	77
P-9600FP Modem	Prentice Corp. 266 Caspian Dr. PO Box 3544 Sunnyvale, CA 94088-3544 (408) 734-9810	This 9600-bps modem offers four-wire full-duplex synchronous operation over multipoint private lines with a millisecond RTS/CTS delay. It can also operate as a V.29 point-to-point modem and has fallback rates of 7200 and 4800 bps. Cost: $1995.	78
TokenPlus Network system	3Com Corp. 1365 Shorebird Way PO Box 7390 Mountain View, CA 94039 (415) 961-9602	A complete line of Token Ring networking products that connects PCs, workstations, and 3Com 3Servers to a 3Com or IBM Token Ring network. Consists of a cabling system and attachments, adapter card, expansion card, operating system, and connect module for the Ethernet and Token Ring networks. Cost: $3755 and up.	79
V.33 Modem	Universal Data Systems 5000 Bradford Dr. Huntsville, AL 35805-1953 (205) 837-8100	This CCITT V.33-compatible modem transmits full-duplex data at 14,400 bps over four-wire private lines, with fallback rates of 12,000 and 9600 bps. Includes Trellis-coded moldulation and automatic adaptive equalization and diagnostic circuitry. Price not listed.	80

IEEE COMPUTER SOCIETY

Membership Application

THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS INC

Schedule of Fees

Membership dues and publication subscriptions are annualized to December 31. Depending upon the date your application is received by the Computer Society, pay the full annual amount or one-half the annual amount as indicated below. Membership expires December 31.

	Half-Year Mar 1–Aug 31	Full-Year Sept 1–Feb 28
1 I don't belong to the IEEE and I want to join just the Computer Society	☐ $19.50	☐ $39.00
2 I already belong to the IEEE and I want to join the Computer Society	☐ $ 7.50	☐ $15.00

IEEE Member Number ______________________

3 I don't belong to the IEEE and I want to join both the Computer Society and the IEEE*

Full IEEE members receive *Spectrum*, *The Institute*, and many additional benefits.

(Total amount to be paid includes annual dues and regional assessment, if any.)

	Half-Year	Full-Year
I reside in Region 1-6 (United States)	☐ $41.00	☐ $82.00
I reside in Region 7 (Canada)	☐ $38.50	☐ $77.00
I reside in Region 8 (Europe, Africa, or the Middle East)	☐ $37.00	☐ $74.00
I reside in Region 9 (Latin America)	☐ $33.50	☐ $67.00
I reside in Region 10 (Asia and Pacific)	☐ $34.00	☐ $68.00

*ACM members who join both IEEE and the Computer Society are entitled to a deduction of $5 off the full-year rate, $2.50 off the half-year rate.

4 In addition to my automatic subscription to *COMPUTER*, I would also like to subscribe to:

	(issues per year)	Half-Year	Full-Year
IEEE Expert (3151)	4	☐ $ 6.00	☐ $12.00
IEEE Computer Graphics and Applications (3061)	12	☐ $13.00	☐ $26.00
IEEE Design and Test (3111)	6	☐ $ 8.50	☐ $17.00
IEEE Micro (3071)	6	☐ $ 8.50	☐ $17.00
IEEE Software (3121)	6	☐ $ 8.00	☐ $16.00
Transactions on Computers (1161)	12	☐ $ 9.00	☐ $18.00
Transactions on Pattern Analysis and Machine Intelligence (1351)	6	☐ $ 7.50	☐ $15.00
Transactions on Software Engineering (1171)	12	☐ $ 9.00	☐ $18.00
Journal on Lightwave Technology (4301)	12	☐ $ 9.00	☐ $18.00
Journal of Robotics and Automation (4401)	4	☐ $ 5.00	☐ $10.00
Journal of Solid State Circuits (4101)	4	☐ $ 4.00	☐ $ 8.00
Proceedings of the IEEE (5011) (available only to IEEE members)	12	☐ $ 8.50	☐ $17.00

☐ Diners Card ☐ VISA ☐ MasterCard ☐ American Express

☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐ Charge Card Number

Mo | Yr ☐☐☐☐ Exp. Date

☐ Check or money order enclosed. Make payable to IEEE

$ ______________ Total amount remitted with this application

Check box for additional information: ☐ Student Membership Application ☐ Publications Catalog

☐ Bulk air option (two-to-six-week delivery of publications) for members in Western Europe, Latin America, and the Far East

I hereby make application for IEEE Computer Society membership and, if elected, will be governed by IEEE's and the Society's Constitutions, Bylaws, and Statements of Policies and Procedures.

MAILING ADDRESS

Full signature ______________________ Date ____________

First Name ____________ Middle initial(s) ____________ Last name ____________

Street address ____________ City ____________ State/Country ____________ Zip ____________

EDUCATION (highest level completed) Course ____________ Degrees received ____________ Date ____________

Name of educational institution ____________ Date of birth ____________ ☐ Male ☐ Female

OCCUPATION ____________ Title or position

Firm name ____________

Firm address ____________

City ____________ State/Country ____________ Zip ____________

Affiliate society membership no. (if applicable) ____________ Grade ____________

REFERENCE (an IEEE member; if unknown, a managerial person who knows you professionally)

Name (print in full) ____________ IEEE Member No. (if applicable) ____________

Position ____________

Street address ____________

City ____________ State/Country ____________ Zip ____________

MAIL TO: IEEE COMPUTER SOCIETY, 10662 LOS VAQUEROS CIRCLE, LOS ALAMITOS, CA 90720–2578

PC1086

Recent IC Announcements

MODEL AND FUNCTION	COMPANY	COMMENTS	R.S. NO.
AH8304TM DAC	Analogic 8 Centennial Dr. Peabody, MA 01961 (617) 246-0300	A video DAC containing a 32-color lookup-table memory and 4-bit tricolor composite video subsystem. Accepts TTL data at update rates to 20 MHz and generates an RS170/RS330/RS343 composite video signal for direct connection to a color monitor. Comes in a 24-pin DIP. Cost: (100's) $76.	85
Bt458 RAMDAC	Brooktree 9950 Barnes Canyon Rd. San Diego, CA 92121 (619) 452-7580	Integrates three 8-bit video DACs and a dual-ported 256 × 24 color palette RAM on a monolithic chip. Pin compatible with the Bt451. Operates at speeds up to 125 MHz. Features an overlay palette, separate frame buffer and microprocessor interfaces, a 60-MHz refresh rate, and ECL-free architecture. Comes in an 84-pin PGA. Cost: (100's) $300 for the 125-MHz Bt458KC125; $265 for the 100-MHz Bt458KC100; and $23 for the 75-MHz Bt458KC75.	86
HE8000 Bipolar gate array	Honeywell Digital Product Center 1150 E. Cheyenne Mtn. Blvd. Colorado Springs, CO 80906 (800) 328-5111 ext. 3414	A 210-ps, 8000-equivalent gate density VLSI monolithic 1C built using Honeywell's ADB-111 1.25-micron bipolar fabrication process. Composed of an uncommitted array of current mode logic gates with 188 ECL 10K/KH, ECL 100K, or CML compatible 1/O cells. Available in 235-pin PGA. Cost: $120,000 for engineering.	87
HS7542 DAC	Hybrid Systems 22 Linnell Circle Billerica, MA 01821 (617) 667-8700	A monolithic 12-bit CMOS 4-quadrant multiplying DAC designed for direct interface to 4- and 8-bit microprocessors. Consists of three 4-bit registers, a 12-bit DAC register, address decoding logic, and a 12-bit CMOS multiplying DAC. Comes in a 16-pin plastic or ceramic D1P. Processing to Mil-Std-883C available. Cost: ranges from $13.95 to $47.15 (100's).	88
OP-400 Op amp	Precision Monolithics PO Box 58020 Santa Clara, CA 95052-8020 (408) 727-9222	A precision quad op amp. Meets PM1's standard OP-07. Maximum supply current is 2.9 mA. Voltage gain exceeds 5 million. Input offset voltage is 150 μV maximum, with a drift of 1.2 μV per degree C. Available in a 14-pin CERD1P. Cost: (in 100's) $5.75 for industrial grade, $17.90 for military grade, and $23.90 for military 883C grade.	89
SS1 K222 Modem 1C	Silicon Systems 14351 Myford Rd. Tustin, CA 92680 (714) 731-7110	Integrates both the US Bell 212A/103 and CC1TT V.22/V.21 1200/300 bps standards. A member of the K-Series, plugs into the K212 slot for upgrade. Features automatic software control to 300 and 600 bps, configuration as a microprocessor peripheral, and a range of 45 db. Cost: (in 100's) $38.85.	90
SS1 207, SS1 6080 Receiver, ADC	Silicon Systems 14351 Myford Rd. Tustin, CA 92680 (714) 731-7110	The SS1 207 integrated monolithic multifrequency receiver detects 15 tone pairs, including ST and KP, and meets Bell and CC1TT R1 central office register signaling specifications. Comes in a 20-pin plastic DIP. Cost: $40.50 in production quantities. The SS1 6080 is a microprocessor compatible successive approximation A/D converter in the commercial and military temperature ranges. Comes in a 22-pin D1P. Cost: under $11 in production quantities.	91
MK41H68 Static RAM	Thomson Components 1310 Electronics Dr. Carrollton, TX 75006 (214) 466-6000	The leading device in a family of CMOS SRAMs. A 4K by 4 memory with a 20-pin industry standard pinout. Available in a version that operates with the LV-TTL JEDEC Standard 3.3V supply. Cost: (in 100's) $34.30 for the 20 ns device ceramic package, $21.45 for the 25 ns part, and $17.85 for the 35 ns part.	92
8023 EEPROM	White Technology 4246 E. Wood St. Phoenix, AZ 85040 (602) 437-1520	A 262,144-bit programmable CMOS EEPROM in 32K-bit by 8-bit/word format. Operates from −155 to +200 degrees C. Common data input and output. Features three-state output. Also available in military and commercial temperature ranges. Comes in a 28-pin CERDIP. Cost: (100's) $588.40.	93
Z84C0006PEC Z80 CPU	Zilog 1315 Dell Ave. Campbell, CA 95008 (408) 370-8000	A 6 MHz CMOS Z80 CPU in the extended commercial temperature range of −40 to +85 degrees C. Also available in the 0 to 70 degrees C range. Comes in plastic packages. Cost: (100's) $5 for the extended commercial range and $3.36 for the other range.	94

CAREER OPPORTUNITIES

RATES: $10.00 per line, $100 minimum charge (up to ten lines). Average six typeset words per line, nine lines per column inch. Add $8 for box number. Send copy at least six weeks prior to month of publication to: **Carole L. Porter, Classified Advertising, COMPUTER Magazine, 10662 Los Vaqueros Circle, Los Alamitos, CA 90720.**

In order to conform to the Age Discrimination in Employment Act and to discourage age discrimination, *COMPUTER* may reject any advertisement containing any of these phrases or similar ones: "...recent college grads...," "...1-4 years maximum experience...," "...up to 5 years experience...," or "...10 years maximum experience." *COMPUTER* reserves the right to append to any advertisement, without specific notice to the advertiser, "Experience ranges are suggested minimum requirements, not maximums." *COMPUTER* assumes that, since advertisers have been notified of this policy in advance, they agree that any experience requirements, whether stated as ranges or otherwise, will be construed by the reader as minimum requirements only.

WORCESTER POLYTECHNIC INSTITUTE

The Computer Science Department invites applications for tenure track faculty positions at all levels from candidates in all areas of specialization. Candidates should have a Ph.D. in Computer Science and a strong interest in both research and teaching.

Worcester Polytechnic Institute emphasizes quality in the undergraduate learning experience and is committed to an innovative project-oriented teaching environment. The quality of the undergraduate computer science degree has been recognized by the recent granting of accreditation by the Computing Sciences Accreditation Board.

The current goal of the Institute is to enhance our graduate program and improve research activities. The department seeks qualified candidates who will help us achieve these objectives. WPI is located close to the center of Massachusetts' minicomputer industry, and excellent opportunities exist for cooperative research and consulting.

The Department has 12 full-time faculty with 200 undergraduates and 35 graduate students in our M.S. and Ph.D. programs. Department equipment includes three VAX 750's, an MV8000, twenty PDP-11's, and 40 PC's. Much of this equipment is networked via two ethernet cables and is also connected to other campus facilities which include a DEC 2060 and numerous VAX's. The Institute is committed to a new Information Science building and full campus networking facilities in the near future.

Located only 45 miles from Boston, Worcester is a small city of 180,000 which has recently undergone a renaissance. It has eight colleges and universities and a rich variety of cultural activities.

Please send a resume to Prof. R.E. Kinicki, Head, Department of Computer Science, WPI, Worcester, MA 01609.

WPI is an Equal Opportunity/Affirmative Action Employer.

TEXAS A & M UNIVERSITY
Department Head Computer Science

Texas A&M University invites applications and nominations for the position of Head of the Department of Computer Science in the College of Engineering. The Department of Computer Science conducts an educational and research program embracing major technical specializations of the profession. The Department offers degrees at bachelors, masters, and doctoral levels and currently has approximately 1,200 students of which 150 are graduate students.

The candidate should hold a Ph.D. degree or equivalent in computer science, computer engineering, electrical engineering, or a related field and be nationally recognized in his or her field. The Head also holds the rank of Professor of Computer Science with tenure. Salary is negotiable with appointment to commence on or before September 1, 1987. The closing for receipt of applications is December 1, 1986. All applications received by this time are assured consideration, but the search will continue until the position is filled. Applicants should send a vita and the names, addresses, and phone numbers of five references to Professor Walter Haisler, Chairman, Computer Science Head Search Committee, Department of Aerospace Engineering, Texas A&M University, College Station, TX 77843. Phone (409) 845-7541.

Texas A&M University is an equal opportunity, affirmative action employer.

PURDUE UNIVERSITY

The School of Electrical Engineering at Purdue University invites applications for faculty positions from qualified individuals who have a strong commitment to teaching and research. Qualifications include an outstanding academic record, a significant involvement with a truly contributive research activity, and a doctorate in electrical or computer engineering. Areas of research specialization which are of particular interest are computers, robotics, and microelectronics. Resumes should be directed to: Head, School of Electrical Engineering, Purdue University, West Lafayette, IN 47907. Immigration status of non-U.S. citizens **must** be stated in application. Purdue is an Equal Opportunity/Affirmative Action employer.

TULANE

The Department of Computer Science invites applications for faculty positions beginning in January/August, 1987.

Tulane is a major private university, selective in enrollment and comprehensive in scope, with about 10,000 undergraduate and graduate students. The campus is located in a residential area of uptown New Orleans, one of America's most distinctive cities. Benefits include dependent tuition, mortgage assistance, relocation expenses, 12% TIAA and typical insurance benefits. The Department of Computer Science, which offers degrees at all levels through the Ph.D., has been specially targeted within the University for growth and support.

Departmental computers systems include a VAX 11/780 with 8 MB main memory running under VMS, a Pyramid 90x with 4 MB high speed main memory running under Unix, a Sun network, numerous LSI-11's, and several microcomputer clusters. The Pyramid is dedicated to departmental research; other systems are divided between research and teaching. In addition to departmental resources, the University maintains an IBM 3081.

Candidates should have a Ph.D. in Computer Science or a related area. Research excellence or demonstrated research potential and a commitment to quality teaching are required.

Applications should be directed to Dr. Johnette Hassell, Head, Department of Computer Science, School of Engineering, Tulane University, New Orleans, LA 70118. Applications must be received by December 1, 1986 for January appointment or March 15, 1987 for August appointment. Tulane University is an Affirmative Action Equal Opportunity Employer.

COMPUTER SYSTEM ANALYST

System analysis. Application and design & implementation. Programming & maintenance of large commercial systems. Management of small software teams. Training & telephone support to clients. Design & programming of new systems. BSC Computer sciences. 4 years experience. $2920 per month. Job site/Interviews Northridge, CA. Send this ad and your resume stating your qualifications to Job # MA 5929, P.O. Box 9560, Sacramento, CA 95823-0560 not later than 10/31/1986.

CHAIR, DIVISION OF COMPUTER SCIENCE
University of California, Davis

The College of Engineering, University of California, Davis campus, is seeking a senior faculty member to lead its new Division of Computer Science to eminence. The Division has been allocated, initially, 15 full-time equivalent faculty positions. We are recruiting for several positions at all levels. At least two of the open positions can be filled with senior level faculty who have outstanding qualifications in the field of computer science.

Qualifications for this position include an outstanding record of research in an area of computer science, demonstrated teaching ability, sensitivity to the needs of a complex University campus, and a potential for leading the faculty to develop excellent research and instruction programs. Applicants should send a resume and letter describing their interests to:

Associate Dean Ray B. Krone
College of Engineering
University of California
Davis, CA 95616

The University of California is an Equal Opportunity/Affirmative Action employer. The position is open until filled.

TENURE-TRACK
FACULTY POSITIONS
GEORGE WASHINGTON UNIVERSITY

The Department of Electrical Engineering and Computer Science invites applications for tenure-track faculty positions. We particularly seek persons active in the areas of Computer Science and Communications. Candidates should have an earned doctorate and research experience, with an interest in both teaching and research. Ability to attract research grants and contracts for support of faculty and student assistants is valued. The George Washington University is located in the center of Washington, DC. This metropolitan area sustains the second largest concentration of research and development activity in the United States which creates a continuing demand for rigorously trained engineers as well as broad research opportunities. Our department is the largest department in the School of Engineering and Applied Science with 33 full-time faculty, and a large graduate and undergraduate student body. The department has an annual research budget of close to $1 million. Send Curriculum Vitae, list of publications and references, to:

Professor Roger H. Lang, Chairman
Department of Electrical Engineering and
Computer Science
School of Engineering & Applied Science
THE GEORGE WASHINGTON UNIVERSITY
Washington, DC 20052

The University is an affirmative action/equal opportunity employer.

UNIVERSITY OF CALIFORNIA AT DAVIS

The Computer Science Division of the University of California et Davis invites applications from highly qualified individuals for several tenure-track positions. The Division is especially interested in candidates whose research is in the fields of computer architecture, networks and distributed systems, artificial intelligence, programming languages/compilers, and VLSI design. Outstanding candidates in other areas, except numerical analysis, will also be considered. Rank and salary will be commensurate with qualifications. All applicants should have a Ph.D. in Computer Science or Computer Engineering. New or recent graduates should show evidence of outstanding research potential, while more senior applicants are expected to have a significant and active research record.

The Division of Computer Science was established three years ago as an independent unit within the Department of Electrical and Computer Engineering. Its interests cover the mainstream areas of computer science, including artificial intelligence, computer systems design, database systems, computer graphics, feult tolerant computers, networking, numerical analysis, programming languages, operating systems, performance evaluation, robotics, software engineering and theoretical computer science.

The Division is in a period of building and growth with the aim of becoming one of the leading computer science departments in the nation. Its expanding departmental research facilities include e VAX 8600, VAX 11-785, several VAX 11-750s, a Ridge 32, Several Sun, Microvax II and Apollo work stations, and a number of special purpose systems all linked through ETHERNET and serial communications networks.

General purpose campus facilities are available, and the supercomputer resources of the Lawrence Livermore National Laboratories are also accessible. Current research strengths and facilities exist in the areas of image processing, computer graphics, computer systems research, microprocessors and robotics.

UC Davis is the third largest of the University of California campuses. It shares with the other campuses of the UC system a strong commitment to leadership in education and research. Davis is close to the major areas of computer science and electronics activity in northern California and thus is in a strategic position to play a key role in future developments of the discipline.

The town of Davis has a population of approximately 50,000. Located eleven miles from Sacramento, it is situated within a short driving distance from the major cultural centers of the San Francisco area as well as the outstanding recreational facilities of the Sierra Nevada Mountains. The town has earned an international reputation for its energy conservation and progressive growth policies. Because of its location, its climate, and its sophisticated small town character, Davis is considered by many an ideal living environment.

Interested applicants should send a curriculum vitae with the names of at least three professional references to Professor Richard F. Walters, Division of Computer Science, or to Professor Lou Hakimi, Department of Electrical and Computer Engineering, University of California, Davis, CA 95616

The University of California is an equal opportunity/affirmative action employer.

ASSISTANT PROFESSOR
DEPARTMENT OF COMPUTER SCIENCE
TRINITY UNIVERSITY
SAN ANTONIO, TEXAS

The Department of Computer Science of Trinity University invites applications for a tenure-track position at the rank of Assistant Professor. A Ph.D. in Computer Science or related discipline is required. The starting date is August, 1987. Salary is competitive and negotiable, and is commensurate with background and experience.

The Assistant Professor will teach 9 hours per semester at the undergraduate level, conduct research, advise students, and serve on appropriate committees.

Trinity University is an independent institution with stable financial resources. Trinity is committed to excellence in the liberal arts and sciences and in selected professional programs including Computer Science. The Department of Computer Science has nearly 125 majors at the undergraduate level. Significant capital expenditures for computing hardware and software indicate the university's commitment to computing for the future. We especially encourage applications from female and minority candidates. Applications should be received by November 15, 1986 and should be mailed to:

Dr. Maurice L. Eggen, Chair
Department of Computing and Information
Sciences
Trinity University
715 Stadium Drive
San Antonio, TX 78284
(512) 736-7480

Trinity University is an Equal Opportunity/Affirmative Action Employer

UNIVERSITY OF SOUTH FLORIDA
Computer Science and Engineering

The Department of Computer Science and Engineering invites applications for faculty positions at the following levels: Instructor, Assistant Professors, Associate Professors and Full Professor. The University of South Florida is the second largest university in Florida, with a current enrollment of over 25,000. It is located on a 1,700 acre campus at the northeast edge of the sunny megatrend city of Tampa. The Tampa metropoliten area is one of the largest and fastest growing in Florida, offering a wide variety of cultural, entertainment, and sports activities. The cost of living is moderate and the quality of life is excellent. The area surrounding the University is experiencing dramatic growth in industry and medical facilities. A variety of firms are located near the campus, including Honeywell, Sperry, IBM, GTE Data Systems, E-Systems, and AT&T.

The Department of Computer Science and Engineering is approximetely 6 years old. It offers Bachelors, Masters and Ph.D. degrees in both Computer Science and Computer Engineering. The department will be housed in a new building in the Fall of 1986, and the size of the faculty is scheduled for rapid expansion over the next three years. A million-dollar endowed chair position was recently announced. Applicants for positions at all levels are especially invited in the areas of Computer Architecture, Distributed Computing/Networking, Computer Vision/Graphics/Image Processing, Artificial Intelligence, and Software Engineering. Exceptional applicants in other areas are welcome.

Applicants should send a resume and the names of three references to: Faculty Search Committee, Computer Science and Engineering, University of South Florida, Tampa, FL 33620. USF is an equal opportunity and affirmative action employer.

GEORGIA INSTITUTE OF TECHNOLOGY

The School of Information and Computer Science invites applications for faculty positions at all levels. Applicants should have a commitment to teaching and should show a record of outstanding research accomplishments or expect to receive a Ph.D. degree by Fall 1987 and show high potential for research. The School seeks applicants to strengthen its capabilities in all areas of current research activity, especially artificial intelligence, software engineering, distributed computer systems, and also in computer graphics, programming languages, theoretical computer science, human factors, VLSI systems, data communications and computer networks, database systems, and computer architecture. Very competitive salaries are offered.

The School is well supported by the Institute. It has 24 faculty members and is currently experiencing substantial faculty growth. Its educational activities include an undergraduate program accredited by the Computing Sciences Accreditation Board, Inc., a Masters program with 150 students and a Ph.D. program with over 70 students. Well equipped separate laboratories support research and education. High-speed local area networks interconnect all major campus laboratories and provide access to CSNet and other national networks.

Georgia Tech is located in Atlanta, which experiences a mild sunbelt climate. It is the center of commerce in the Southeast, offering a diverse economy and good employment opportunities in all professional areas. Atlanta offers good cultural and recreational opportunities, extremely attractive residential neighborhoods, and affordable housing.

Candidates should send complete resumes and names of at least three references to: Professor Nancy D. Griffeth, Chairman, Faculty Search Committee; School of Information and Computer Science; Georgia Institute of Technology; Atlanta, Georgia 30332.

Georgia Tech is an equal opportunity employer.

NEW MEXICO STATE UNIVERSITY
Department of Computer Science

Applications and nominations are invited for the position of Academic Department Head of the Department of Computer Science at New Mexico State University. The Department was established in 1967, making it one of the earliest computer science departments. Growth potential is great, since in 1983 the State of New Mexico established the Computing Research Laboratory (CRL) at New Mexico State University as one of five centers of technical excellence in the state. To realize this potential, we are seeking a person with leadership capability, a strong record of research, solid academic credentials, and good administrative and interpersonal skills. Major responsibilities include aggressive recruiting of new faculty, leading the development of research programs, and enhancing computing facilities. Qualifications for the position include a Ph.D. in Computer Science or a closely-related area. This is a senior level, tenure-track position with competitive salary. The search committee will start considering applications on November 1, 1986 and the position will be filled as soon as an acceptable candidate is identified; the starting date could be as early as January 1, 1987.

New Mexico State University is a land grant institution with an enrollment of 13,000 students. The Department of Computer Science is in the College of Arts and Sciences and offers B.S., M.S., and Ph.D. degrees. There are presently 250 undergraduate majors, 45 graduate students in the M.S. program and 23 in the Ph.D. program, and 13 full-time faculty members in addition to the Department Head. Faculty research interests include algorithms, artificial intelligence, computer architecture, computer graphics, computer vision, computer-aided education, database systems, and programming languages.

There is close cooperation between the Department of Computer Science and the CRL center of technical excellence. The CRL has its own professional staff including faculty, visiting scientists, post-doctoral students, and graduate students, and pursues programs of basic and applied research in such areas as artificial intelligence, distributed systems, knowledge representation, human/computer interactions, machine sensory systems, and knowledge-based control. The CRL offers exciting research opportunities for the Department of Computer Science. A new building, which will house the Department and the CRL, will be completed in spring, 1987 and will offer excellent facilities for both organizations.

Applicants for the position should send a letter of application and a complete resume including the names, addresses, and telephone numbers of at least three references to Professor Yorick Wilks, Department of Computer Science, Box 3CU, New Mexico State University, Las Cruces, NM 88003-0099.

AA/EOE EMPLOYER

PROGRAMMER ANALYST

Analyze business procedures & problems; study existing data handling systems & develop new systems; specify in detail logical operations & prepare programs for computer use. Knowledge of Database design and methodology including TSO/ISPF, IMS-DB/DC, PAN-VALET on-line & ADF II, COBOL & JCL essential. 30 semester (45 quarter) units of college in programming, data processing or computer science & 2 years experience in subject job or Systems Analyst required. A Bachelor's Degree may be substituted for 1 year of experience. $2050/month. Jobsite: San Bernardino, CA. Send this ad & a resume to Job #MA6044, P.O. Box 9560, Sacramento, CA 95823-0560 not later than 10-31-86.

CENTER FOR AUTOMATION RESEARCH

Computer Vision Assistant, Associate and Full Research Scientists to direct advanced research programs in computer vision. One year renewable positions whose availability and continuation are dependent on contract/grant funds. Ph.D. in computer science or closely related discipline required. Experience in computer vision research required, specifically in areas such as three-dimensional and time-varying scene analysis, robot navigation, knowledge-bases vision systems, and computer architectures for vision. Salary up to $60,000 depending on level of position. The closing date for accepting applications is September 1, 1987. Submit signed and dated curriculum vitae plus three references to: Prof. Azriel Rosenfeld, Center for Automation Research, University of Maryland, College Park, MD 20742. AA/EOE.

UNIVERSITY OF UTAH

Computer Science Faculty Positions

The Department of Computer Science at the University of Utah solicits applications at all professional ranks. A candidate for Assistant Professor must earn the PhD in Computer Science or related field prior to September. A candidate for Associate Professor must have, in addition, at least three years of teaching or research experience in Computer Science. A candidate for the rank of Professor must have a well-established research record in Computer Science.

The Department currently has 15 tenure-track faculty members, 2 teaching faculty, as well as an additional 11 serving in research and adjunct capacities. The student population includes approximately 200 undergraduate majors, 40 Master's degree students and 35 PhD students. Substantially funded faculty research projects include: asynchronous computation, computer-aided geometric design: computer-aided instruction, computer graphics, computer system architecture, computer vision, models of computation, program verification, programming languages, robotics, sensory information processing, software for small computer systems, symbolic and algebraic computation, theory of computation, and very large scale integrated circuit design. Funding sources include major support from the NSF/CER program and from DARPA. Other governmental agencies and private industry also provide significant support.

Over the past few years, the department has created an outstanding research environment. The department maintains a professionally staffed research computing facility which includes a DEC system 2060, VAX 8600, five other VAX mainframes and a Gould 9080. An 18-node BBN Butterfly parallel computer system and over 70 Hewlett-Packard, Apollo and Sun workstations are installed. The facility is connected to most major geographic networks (Arpanet, CSNET, USENET and Telenet). A computer graphics laboratory with equipment representative of the most advanced devices available in the industry is also available. Other specialized laboratories include facilities for image processing and understanding, robotics, parallel processing, Lisp programming, computer aided instruction and VLSI research.

Starting date for appointment is July 1987. Direct vita, along with the names of three or more references, by March 1, 1987 or until positions are filled, to:

Richard F. Riesenfeld
Recruiting Committee Chairman
Department of Computer Science
University of Utah
Salt Lake City, Utah 84112

The University of Utah is an Affirmative Action/Equal Opportunity Employer and especially encourages applications from women and members of minority groups.

THE UNIVERSITY OF TEXAS AT ARLINGTON

The Department of Computer Science Engineering (CSE) at the University of Texas at Arlington (UTA) invites your application for tenure-track or visiting faculty positions in all areas of computer science and computer engineering. Rank is open. An earned doctorate or equivalent and commitment to teaching and scholarly research is required. Openings are expected for January, June, or September 1987. Our department offers baccalaureate, masters, and doctoral programs. Excellent computing facilities are available including personal computers in each CSE faculty office. UTA is in the Dallas/Fort Worth metropolitan area near a large concentration of high technology industries. Interested persons should send a resume to Professor Bill D. Carroll, Professor and Chairman, Computer Science Engineering Department, P.O. Box 19015, The University of Texas at Arlington, Arlington, TX 76019. Phone 817-273-3785.

The University of Texas at Arlington is an Equal Opportunity Affirmative Action Employer.

CASE INSTITUTE OF TECHNOLOGY

Case Western Reserve University

We invite applications for tenure-track positions at all levels. We are particularly interested in candidates whose research areas include VLSI systems and design automation, applied artificial intelligence, data bases, software design environments, and analysis of algorithms. Applicants will be judged primarily on their ability to strengthen our quest for excellence in research and teaching.

CWRU is a small private university with a total enrollment of about 8400, of which about 5100 are graduate and professional students. The university campus is the hub of the pleasant area known as University Circle, an incorporation with neighboring cultural centers and museums. University Circle is about 10 miles from downtown Cleveland.

Case Institute of Technology, a subunit of CWRU, is among the top ten engineering schools in terms of research funding per faculty member and undergraduate student quality. The department of Computer Engineering and Science has a young faculty of 10 and growing, and a graduate student body of 115 students, 48 of which are currently in the Ph.D. program. Department facilities include a DEC VAX-11/780, Data General MV/10000, twelve desk-top computers (Intel, DEC and Apollo), several color graphic displays (four high and ten medium resolution), and hard-copy equipment (color ink jet and laser printers, plotters, etc.). Faculty and students participating in the Center for Automation and Intelligent Systems Research have access to the Center's VAX-11/782 and VAXstation 100 display systems. Educational computing is provided by the Computing Center with three DEC-2060's, the Case Personal Computer Lab with 48 DEC PRO 350's, four AT&T 3B2's and several AT&T 6300's, and the department's MV/10000.

Applicants should submit their resume and names of three references to: Professor J. Thomas Mortimer, Chairman; Department of Computer Engineering and Science; Case Western Reserve University, Cleveland, Ohio 44106.

An Equal Opportunity Affirmative Action Employer.

SYSTEM ANALYST

Analyze data-processing requirements to determine data-processing system. Provide capabilities required by customers. Plan & layout new system, modify existing system. Knowledge of IBM 4381/4341 mainframes, DOS/VSE, JCL, VM/CMS, CICS, IBM 8100, Data Base IDMS, COBOL. BSc in Mathematics of Computer Science. 3 years experience. $2920 per month. Send this ad and your resume to Job #DC-5880, P.O. Box 9560, Sacramento, CA 95823-0560, not later than 10-31-86.

THE UNIVERSITY OF ALBERTA
Department of Computing Science

The Department of Computing Science is undergoing an extensive expansion in research initiatives. Applications are invited for two tenure-track positions at the Assistant/Associate Professor level. Responsibilities include research as well as teaching at the graduate and undergraduate levels. Candidates from all areas will be considered. Current hardware support includes an Amdahl 5870, a network of VAX 11/780's, and well equipped microcomputer and workstation laboratories for graphics, VLSI, and AI research. Access to a Cyber 205 is available. Salary range is $31,612 to $50,630 and is commensurate with qualifications and experience. Send curriculum vitae, names of three references, and up to three reprints or papers. New Ph.D.'s should also include a copy of their transcript. Apply to: Dr. Lee White, Chairman, Department of Computing Science, University of Alberta, Edmonton, Alberta, T6G 2H1. Applications will be accepted until December 31, 1986. The University of Alberta is an equal opportunity employer.

MICHIGAN STATE UNIVERSITY

The Department of Computer Science invites applications for tenure track positions at all levels. Candidates from all areas of specialization in computer science or computer engineering will be considered. The department has a special interest in candidates in the areas of programming languages, database systems, artificial intelligence and expert systems, robotics, design of computer systems and networks, parallel computation, dataflow machines, operating systems and computational complexity. Candidates should have a Ph.D. in computer science or computer engineering and have a strong interest in both research and teaching. Applications will be accepted until the positions are filled.

As a unit within the College of Engineering at Michigan State University, Computer Science offers the Bachelor of Science, Master of Science and Doctor of Philosophy degrees. Special support is available from within the college and university to initiate research by new faculty members. Michigan State University enjoys a park-like campus of 2,100 developed acres and 3,100 acres of experimental farms, outlying research facilities and natural areas. The campus is adjacent to the cities of East Lansing and the capital city, Lansing. The Greater Lansing area has approximately 250,000 residents. The communities have fine school systems and place a high value on education.

Applicants should send a resume and a statement of research and teaching interests to:

Dr. Anthony S. Wojcik, Chairperson
Department of Computer Science
A714 Wells Hall
Michigan State University
East Lansing, Michigan 48824-1027
CSNET: wojcik @ mich-state

Michigan State University is an Equal Opportunity/Affirmative Action Institution and encourages applications from members of ethnic minority groups.

UNIVERSITY OF PENNSYLVANIA SCHOOL OF MEDICINE
Executive Director for Computing and Instructional Technology

The University of Pennsylvania is currently seeking nominations and applications for the new position to direct the development of information management systems in the prestigious School of Medicine. We offer a competitive salary and comprehensive fringe benefits including liberal paid time-off, extensive tuition reimbursement, excellent retirement programs, use of athletic facilities and a convenient location in a beautifully landscaped urban environment.

DUTIES: Refine and implement the School's strategic computer plan. Plan the installation and operation of School-wide service facilities, including local area networks. Act as liaison with central computing and hospitals affiliated with the school. Facilitate and provide support services for academic and administrative computing needs. Review and coordinate all computer hardware and software and services acquisitions.

QUALIFICATIONS: Graduate degree with appropriate computing and managerial experience. Academic appointment will be considered for suitable candidate. Experience in a university environment directing a computing facility or responsible for extensive systems development preferred. Familiarity with long-range computer planning and broad knowledge of current data processing technology.

EXPRESS YOUR INTEREST by submitting your resume and salary requirements by October 15, 1986 to Linda Frank, University of Pennsylvania, 3451 Walnut Street, Philadelphia, PA 19104-6286. An equal opportunity/affirmative action employer.

UNIVERSITY OF CALIFORNIA SANTA BARBARA

The UNIVERSITY OF CALIFORNIA AT SANTA BARBARA invites applications for four tenure track positions at senior professional levels in the Department of Computer Science, and for three additional positions in closely related areas, including computer engineering. The Department of Computer Science is in a rapidly expanding College of Engineering that has achieved national prominence in many areas of research. UCSB and the College of Engineering are highly committed to augmenting its Computer Science Department into a department that has both excellence and national visibility. The senior positions in Computer Science currently available are in three major areas of research in which the Department of Computer Science intends to achieve its greatest strength. These areas are: machine vision and motion planning; parallel computing; and software engineering. These positions will provide opportunities for successful candidates to guide the growth and development of the Department in these areas. Strong material support will be given to such positions. "State of the Art" laboratories are currently being planned for instructional purposes in these areas. Research facilities in these areas will be made available and tailored to the needs of successful applicants. Graduate student support will be given high priority.

Interactions with various research groups on campus (Center for Robotic Systems, Cognitive Science Program, Center for Scientific Computing, Institute for Theoretical Physics, etc.) will be strongly encouraged and supported.

Applicants must possess a doctoral degree and an extremely strong record of research accomplishments. Teaching experience is desirable. Outstanding younger candidates are also encouraged to apply. Deadline for receipt of applications is January 15,1987. Send resume and names of at least eight references to:

Chairman
Computer Science Faculty Search
Committee
Department of Computer Science
University of California
Santa Barbara, CA 93106

The University of California is an Equal Opportunity/Affirmative Action Employer.

OHIO STATE UNIVERSITY
Department of Computer and Information Science

Applications are invited for faculty positions at all levels. A Ph.D. in computer science or a closely related field is required. Of special interest are candidates in the areas of artificial intelligence, computer graphics, data bases, parallel and distributed computing, software engineering, and VLSI design. Special research support packages are negotiable for highly qualified candidates.

Research computing facilities within the Department include DEC 20/60, VAX 11/780, IBM 4341, AT&T 3B2, Xerox Lisp and XDE machines, Sun and HP workstations, and several experimental multi-computers (Intel Hypercube, BBN Butterfly, etc.). In addition to the University's computing facilities (IBM 3081-D, etc.), the Department also has access to national networks including ARPANET and CSNET.

To apply, please send application and resume, including a statement of research and teaching interests and the names and addresses of at least three references, to Prof. Ming T. (Mike) Liu, Chairman of Faculty Search Committee, Department of Computer and Information Science, The Ohio State University, 2036 Neil Avenue, Columbus, Ohio 43210-1277. (CSNET/ARPANET: LIU @ Ohio-State.)

The Ohio State University is an opportunity/affirmative action employer.

OHIO STATE UNIVERSITY
Department of Computer and Information Science

Applications are invited for faculty positions at all levels. A Ph.D. in computer science or a closely related field is required. Of special interest are candidates in the areas of artificial intelligence, computer graphics, data bases, parallel and distributed computing, software engineering, and VLSI design. Special research support packages are negotiable for highly qualified candidates.

Research computing facilities within the Department include DEC 20/60, VAX 11/780, IBM 4341, AT&T 3B2, Xerox Lisp and XDE machines, Sun and HP workstations, and several experimental multi-computers (Intel Hypercube, BBN Butterfly, etc.). In addition to the University's computing facilities (IBM 3081-D, etc.), the Department also has access to national networks including ARPANET and CSNET.

To apply, please send application and resume, including a statement of research and teaching interests and the names and addresses of at least three references, to Prof. Ming T. (Mike) Liu, Chairman of Faculty Search Committee, Department of Computer and Information Science, The Ohio State University, 2036 Neil Avenue, Columbus, Ohio 43210-1277. (CSNET/ARPANET: LIU @ Ohio-State.)

The Ohio State University is an equal opportunity/affirmative action employer.

POSTDOCTORAL RESEARCH POSITION

Electrical Engineering Department, University of Southern California. Temporary full time position for one to three years commencing November 1, 1986 with possibility of extension contingent upon funding. Salary $44,000 to $50,000 per annum, dependent on qualifications. Assist in development of a knowledge based system for design of testable systems with professors and graduate students. Ph.D. degree or equivalent in Computer Science or Computer Engineering required, and research experience in testing, design for test, fault tolerant systems, expert systems and VLSI design. Send resume to Professor M.A. Breuer by October 30, 1986, Electrical Engineering-Systems Dept. SAL 326, University of Southern California, Los Angeles, CA 90089-0781. USC is an equal opportunity, affirmative action employer.

SOUTHERN ILLINOIS UNIVERSITY AT CARBONDALE
Senior/Junior Faculty Positions in the Computer Science Department

Applications are invited for four full time tenure track faculty positions in a growing, dynamic department, beginning August 15, 1987.

1. One position at the professional level: Candidates should have an established and substantial research record evidenced by scholarly publications and external funding. The applicant would be expected to provide leadership in a newly developing doctoral program along with excellence in teaching. Applications will be accepted until December 31, 1986 or until the position is filled.
2. One position at the associate professor level: Applicants should have an established research record in one of the mainstream areas of Computer Science, and would be expected to participate fully in the department's graduate program and provide excellence in teaching. Applications will be accepted until November 15, 1986 or until the position is filled.
3. Two positions at the assistant/associate professor level: Candidates for these positions should demonstrate promise of ongoing and future research as well as a commitment to teaching. Applications in all areas of Computer Science will be considered, with the department specifically seeking expertise in the areas of programming languages and operating systems. Applications will be accepted until November 30, 1986 or until the positions are filled.

Resumes should be sent to: Faculty Recruitment Committee, Department of Computer Science, Southern Illinois University, Carbondale, IL 62901. SIUC is an Equal Opportunity, Affirmative Action Employer.

VISITING PROFESSORSHIPS & VISITING RESEARCH PROFESSORSHIPS
School of Engineering and Applied Science
The George Washington University

Visiting Professorships at junior and senior levels are available in specialty areas that include, but are not limited to, the following:

- Application Areas of Management
- Artificial Intelligence
- Communications
- Computer Aided Design
- Computer Aided Manufacturing
- Computer Assisted Engineering
- Computer-Integrated Manufacturing
- Computer Science
- Environmental Engineering
- Human Factors
- Information Management
- Management of Research and Development
- Mathematical Optimization
- Mechanical Engineering
- Reliability
- Robotics
- Simulation
- Stochastic Processes
- Structural Engineering
- Systems Analysis and Management
- Water Resources

Appointments are for one-year periods. Applicants should send vita, including complete publication list, and three references to:

Office of the Dean
School of Engineering & Applied Science
THE GEORGE WASHINGTON UNIVERSITY
Washington, CA 20052

The University is an affirmative action and equal opportunity employer.

INFORMATION SYSTEMS

Manager to integrate office and information technologies within company. Design, develop, implement long range information systems planning methodologies to support business strategy. Act as liaison between department heads & management to define nature of proposed projects, determine information requirements, evaluate & select software & hardware needs, ensure standardization within company. Requires BS in computer or system science, or related, plus 4 years minimum experience. Must be familiar with IBM/HP hardware, working knowledge of FORTRAN, Pascal, COBOL, C, CAD/C. Must have good foundation in database management experience in networking & interfacing, plus experience in manufacturing/engineering environment. Salary $45,000/year. Job site/interview: Burbank, CA. Send this ad and your resume to: Box CP-2, 10662 Los Vaqueros Circle, Los Alamitos, CA 90720.

WASHINGTON UNIVERSITY
DEPARTMENT LEADER

The Department of Electrical Engineering at Washington University in Saint Louis is seeking a new Chairman to guide an expansion of about one-third from its present eighteen person level; additional positions are anticipated from retirement over the next decade. A major gift to underwrite the opening phases of this development is already in hand. Persons desiring additional Information should write or telephone Professor Fred J. Rosenbaum, Department of Electrical Engineering, Washington University, Saint Louis, Missouri 63130 [(314)889-6154]. Washington University is always interested in superior applicants regardless of race, creed, color, gender, or age.

University of Utah
Department of Computer Science
Department Chairperson

The Computer Science Department at the University of Utah invites applications for the tenure track position of Chairperson. This appointment is for a three year, renewable term.

The Department has an internationally recognized reputation for excellence in research and teaching, both at the undergraduate and graduate levels. There are currently 15 regular faculty, and the full complement of faculty numbers 28. The student population of the Department comprises approximately 35 PhD students, 40 MS students, and 250 undergraduate majors selected by an annual competition among pre-majors.

Active research areas within the Department include computer aided geometric design, VLSI design, information retrieval architecture, parallel processing, computer simulation, robotics and image processing, portable AI systems software, information based complexity theory, computer aided instruction, functional and logic programming, and computer music.

The Department maintains a superb research computing facility, including a DECSystem 2060 and five VAX mainframes, including a VAX 8600. An 18-node BBN Butterfly and over seventy HP, Apollo, and Sun workstations are installed. Special purpose equipment is available for signal processing, graphics, robotics, and VLSI research.

Starting date for the appointment is July 1, 1987. Direct vita, along with the names of three references, to:

Gary Lindstrom, Chairman
Chairperson Search Committee
University of Utah
Department of Computer Science
Salt Lake City, Utah 84112

CONSIDERATION OF APPLICATIONS WILL CONTINUE UNTIL 1 FEBRUARY 1987 OR UNTIL THE POSITION IS FILLED.

THE UNIVERSITY OF UTAH IS AN AFFIRMATIVE ACTION, EQUAL OPPORTUNITY EMPLOYER, AND WELCOMES APPLICATIONS FROM WOMEN AND MINORITY CANDIDATES.

LET US PLACE YOU IN A BETTER JOB NOW

Put our 20 years of experience placing technical professionals to work for you. Client companies pay all fees, interview and relocation costs. You get our expert advice and counsel FREE. Nationwide opportunities in Communications, Defense, Intelligence, Computer, Satellites, and Aerospace Systems.

We are seeking individuals with experience and interest in one or more of the following areas:

- **Software Configuration Management**
- **Data Base Design and Development**
- **Distributed System Design and Development**
- **VAX Software Development Under VMS**
- **IBM 4341 Software Development**
- **FORTRAN and MACRO Programming**
- **Military Standard Systems Design and Development**
- **Local Area Networks**
- **Color Graphics Display Software Design**
- **Rapid Prototyping**
- **Software Quality Assurance**
- **Artificial Intelligence**
- **Test Planning and Testing**
- **Verification and Validation**

Salaries range from $30,000-$75,000 plus. U.S. citizenship required. EBI/SBI desirable. Let us place you in a better, more rewarding job . . . now. Send your resume in confidence to:
Dept. CA-IC

WALLACH
associates, inc.

Washington Science Center
6101 Executive Boulevard, Box 6016
Rockville, Maryland 20850-0616

Technical and Executive Search

Wallach . . . Your Career Connection

ADVERTISER INDEX

ACM Computer Science Conference '87 70
ACM/IEEE 24th Design Automation Conference 44
ASPLOS-II 101
AT&T 5
The BDM Corp. 9
CACI 1
Central Intelligence Agency 86
Computer Networking Symposium 103
Computer Society Membership 105
Computer Society Ombudsman 96
Digital Equipment Corp. 6-7
FJCC '86 87-89
GTE Government Systems Corp. ....Cover II
ICDCS '87 ....Cover III
IEEE 16th Videoconference 16
IEEE Symposium on Security and Privacy '87 69
Index Technology Corp. ....Cover IV
Int'l Conf. on Entity - Relationship Approach 58-59
Int'l Symposium on Computer Architecture 20
Jackson Chamber of Commerce 97
Language Processors, Inc. 19
The MIT Press 94
National Business Employment Weekly 91
Pecan Software Systems, Inc. 84
Real-Time Systems Symposium '86 28
Sun Microsystems 14-15
University of Petroleum & Minerals 93
University of Utah 111
Wallach Associates, Inc. 111
Wang Institute 81

FOR DISPLAY ADVERTISING INFORMATION CONTACT

Southern California and Mountain States: Richard C. Faust Company, 24050 Madison Street, Suite 100, Torrance, CA 90505; (213) 373-9604.
Northern California and Pacific Northwest: Don Farris Company, 161 W. 25th Ave., #102B, San Mateo, CA 94403; (415) 349-2222.
Jack Vance, P.O. Box 3205, Saratoga, CA 95070; (408) 741-0354.
East Coast: Hart Associates, Inc., P.O. Box 339, 42 Lake Blvd., Matawan, NJ 07747; (201) 583-8500.
New England: Arpin Associates, P.O. Box 227, Weston, MA 02193; (617) 899-5613.
George Watts, III, 4 Conifer Dr., Wilbraham, MA 01095; (413) 596-4747.
Midwest: Thomas Knorr, Knorr MicroMedia, Inc., 333 North Michigan Ave., Chicago, IL 60601; (312) 726-2633.
Southwest: The House Company, 3000 Weslayan, Suite 345, Houston, TX 77027; (713) 622-2868.

Advertising Manager: Mike Koehler, *COMPUTER* Magazine, 10662 Los Vaqueros Circle, Los Alamitos, CA 90720; (714) 821-3240, 821-8380.

For production information, conference or classified advertising contact Carole L. Porter, *COMPUTER* Magazine, 10662 Los Vaqueros Circle, Los Alamitos, CA 90720; (714) 821-1140.

PRODUCT INDEX

	RS#	Page#
Adams-Russell Electronics	40	98
Anadigics	41	98
Analogic	85	106
Apollo Computer Inc.	65	102
Ashton-Tate	66	102
Black Box Corp.	70	104
Borland International, Inc.	69	102
Brooktree	86	106
Bubbl-tec	71	104
Cray Research, Inc.	67	102
Data General	37	100
DEC	68	102
EEsof	60	99
Floating Point Systems	63	100
Force Computers, Inc.	72	104
Ford Microelectronics	42	98
Gain Electronics	43	98
GE Electronics Laboratory	44	98
GigaBit Logic	45	98
Gold Hill Computers	62	100
Harris Microwave Semiconductor	46	98
Heurikon Corp.	73	104
Honeywell	47,87	98,106
Hybrid Systems	88	106
IBM	74	104
Index Technology Corp.	7	Cover IV
ITT GaAs Technology Center	48	98
Jackson Chamber of Commerce	6	97
Language Processors, Inc.	1	19
May-Craft Information Systems	75	104
McDonnell-Douglas Microelectronics	49	99
Metheus Corp.	38	100
Micromotion	76	104
Microwave Associates	50	99
Microwave Semiconductor	51	99
The MIT Press	5	94
National Business Employment Weekly	4	91
National Semiconductor Corp.	77	104
Pacific Monolithics	52	99
Pecan Software Systems, Inc.	3	84
Precision Monolithics	89	106
Prentice Corp.	78	104
RCA Advanced Technology Labs	53	99
Silicon Systems	90,91	106
Tandem Computers, Inc.	64	102
Thomson Components	92	106
3Com Corp.	79	104
TI Defense Systems & Electronics	54	99
TriQuint Semiconductor	55	99
Universal Data Systems	80	104
Varityper	39	100
Vitesse Electronics Corp.	56,61	99
Wang Institute	2	81
White Technology	93	106
Zilog	94	106

COMPUTER READER SERVICE

For further information on advertised products, new products, or literature and to specify levels of interest in *Computer*'s articles and departments fill out one of these cards. Affix stamp and mail.

Please print or type your name and address.

Reader Service Card

COMPUTER

Void after 4/15/87

10/86

Name ______________________

Company ______________________

Title ______________________

Address ______________________

City ______________ State ______________

Zip ______________ Telephone ______________

Circle your level of interest in the following articles and departments:

	High	Medium	Low	Did not read
Guest Editor's Intro	101	102	103	104
DARPA, SDI, and GaAs	105	106	107	108
GaAs Digital ICs	109	110	111	112
Signal Processor	113	114	115	116
Design Issues	117	118	119	120
32-bit RISC	121	122	123	124
RISC Microprocessor	125	126	127	128
Update	129	130	131	132
Open Channel	133	134	135	136
Book Reviews	137	138	139	140
New Products	141	142	143	144
New Literature	145	146	147	148

I need more information on the following:

1	2	3	4	5	6	7	8	9	10	11	12	13	14
15	16	17	18	19	20	21	22	23	24	25	26	27	28
29	30	31	32	33	34	35	36	37	38	39	40	41	42
43	44	45	46	47	48	49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	64	65	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80	81	82	83	84
85	86	87	88	89	90	91	92	93	94	95	96	97	98

Products purchased or specified:

	Hobby	Job
Computers	150	151
Peripherals	152	153
Data Communication	154	155
Memories/components	156	157
Software and Services	158	159
Publications	160	161

170 Please send me the IEEE-CS 1986 Publications catalog.

171 Please send me an application for advancement to senior member.

172 Please send me an IEEE-CS membership application.

Reader Service Card

COMPUTER

Void after 4/15/87

10/86

Name ______________________

Company ______________________

Title ______________________

Address ______________________

City ______________ State ______________

Zip ______________ Telephone ______________

Circle your level of interest in the following articles and departments:

	High	Medium	Low	Did not read
Guest Editor's Intro	101	102	103	104
DARPA, SDI, and GaAs	105	106	107	108
GaAs Digital ICs	109	110	111	112
Signal Processor	113	114	115	116
Design Issues	117	118	119	120
32-bit RISC	121	122	123	124
RISC Microprocessor	125	126	127	128
Update	129	130	131	132
Open Channel	133	134	135	136
Book Reviews	137	138	139	140
New Products	141	142	143	144
New Literature	145	146	147	148

I need more information on the following:

1	2	3	4	5	6	7	8	9	10	11	12	13	14
15	16	17	18	19	20	21	22	23	24	25	26	27	28
29	30	31	32	33	34	35	36	37	38	39	40	41	42
43	44	45	46	47	48	49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	64	65	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80	81	82	83	84
85	86	87	88	89	90	91	92	93	94	95	96	97	98

Products purchased or specified:

	Hobby	Job
Computers	150	151
Peripherals	152	153
Data Communication	154	155
Memories/components	156	157
Software and Services	158	159
Publications	160	161

170 Please send me the IEEE-CS 1986 Publications catalog.

171 Please send me an application for advancement to senior member.

172 Please send me an IEEE-CS membership application.

Comments on articles and other editorial matter

What did you find appealing about the articles/departments you liked? ______________________

____

____

____

____

What other topics/departments would you like to see?

____

____

____

General comments: ______________________

____

____

For reader service, see other side

This PO box for reader service cards only.

PLACE
STAMP
HERE

COMPUTER
magazine

Reader Service Inquiries
Box 24008
Los Angeles, Ca 90024
USA

Comments on articles and other editorial matter

What did you find appealing about the articles/departments you liked? ______________________

____

____

____

____

What other topics/departments would you like to see?

____

____

____

General comments: ______________________

____

____

For reader service, see other side

This PO box for reader service cards only.

PLACE
STAMP
HERE

COMPUTER
magazine

Reader Service Inquiries
Box 24008
Los Angeles, Ca 90024
USA

The 7th International Conference on **Distributed Computing Systems**

BERLIN (West), Germany,
21 – 25th September 1987
International Congress Center
Berlin

call for papers

The Conference

ICDCS 87 is dedicated to various aspects of distributed computing systems, one of the most challenging domains of computer science. The Conference is a milestone for reviewing the state of the art at international level and outlining the future technical developments in distributed systems.

Tutorials will be offered on the first two days of this International Conference and keynote speakers will be invited to address the developments in Europe, America and Asia.

The conference and tutorials will be held at the International Congress Center Berlin.

Historic Berlin, celebrating its 750th anniversary in 1987, will offer an exciting environment for other activities (festivals, exhibitions, entertainment).

Areas of interest

The 7th ICDCS will embrace the theory, design and implementation of distributed systems.
Subject areas include

- decentralized computer architectures
- communication architectures and protocols
- distributed operating systems and system software
- distributed applications
- distributed database systems
- software engineering and languages for distributed systems
- reliability, fault tolerance and security
- modelling and performance evaluation
- theory and formalism

Information for Authors

Authors are requested to submit five copies (in English) of their double-spaced typed manuscript (max. 20 pages) as follows:

If your are in **E**urope or **A**frica send to

G. LE LANN
E/A Program Chairman
INRIA Rocquencourt BP 105
78153 Le Chesnay Cedex,
France
Phone (331) 39635364

if you are in **A**merica (North or South), **A**sia or **A**ustralia send to

K. H. (Kane) Kim
A/A/A Program Chairman
Computer Engineering Program
Dept. EE, Room 544B
University of California, Irvine
Irvine, CA 92717 USA
Phone (714) 856-4821
COMPMAIL: K. KIM

The Conference proceedings will be published by the IEEE Computer Society. Authors of all accepted papers will be expected to sign a copyright release form.

Selected papers of exceptional interest and merit will be considered for publications in one of the major IEEE journals.

In addition to papers, proposals for one day tutorials are solicited in any of the conference topics. Such proposals should be submitted by January 1, 1987 to:

Earl E. Swartzlander, Jr.
TRW Defense Systems Gr.
One Space Park 02/2791
Redondo Beach, CA 90278
USA

or

L. Pouzin
CNET Centre Paris A
38-40 Rue de Gen. Leclerc
9231 Issy le Moulineaux
France

Sponsored by

THE COMPUTER SOCIETY
OF THE IEEE

The Institute of Electrical and Electronics Engineers, Inc.

in cooperation with

Hahn-Meitner-Institut Berlin

Gesellschaft für Informatik e.V.

Conference Organization

General Chairman
H. Weber (FRG)

Honorary Chairman
K. Zander (FRG)

E/A Chairman
R. Popescu-Zeletin (FRG)

A/A/A Chairman
W. C. McDonald (USA)

E/A Program Chairman
G. Le Lann (F)

A/A/A Program Chairman
K. H. (Kane) Kim (USA)

Decentralized Computer Architecture
Vice-Chair: W. D. Carroll (USA)
H. Aiso (Japan), D. W. Mizell (USA), J. Lipovsky (USA), W. Giloi (FRG), J. S. Banino (F), I. M. Barron (U.K.)

Communication Architectures and Protocols
Vice-Chair: A. Danthine (Belgium)
H. Zimmermann (F), I. Chlamtac (Israel), A. Cantoni (Australia), E. Manning (Canada), M.T. Liu (USA), W.W. Chu (USA)

Distributed Operating Systems
Vice-Chair: A. van Tilborg (USA)
L. Svobodova (CH), L. Simoncini (I), J. Altaber (CH), J.T. Ellis (USA), T. Lawrence (USA), E. Lazowska (USA)

Applications
Vice-Chair: N. Naffah (F)
S. Schindler (FRG), M. Sloman (U.K.), P. M. Franca (Brazil), E. Shipley (USA), M. Malek (USA)

Distributed Database Systems
Vice-Chair: H. Garcia-Molina (USA)
G. Gardarin (F), H. J. Schneider (FRG), A. Ceri (I), P. B. Berra (USA), S. K. Rahimi (USA)

Software Engineering, Languages and Tools
Vice-Chair: M. Diaz (F)
T. Maibaum (U.K.), C. Vissers (NL), R. B. Grafton (USA), P. Löhr (FRG), Y. Matsumoto (Japan), C. V. Ramamorthy (USA), S. S. Yau (USA)

Reliability, Fault Tolerance and Security
Vice-Chair: H. Kopetz (A)
J. C. Laprie (F), T. Anderson (U.K.), A. Avizienis (USA), V. Gligor (USA), H. Hecht (USA)

Modelling and Performance Evaluation
Vice-Chair: T. C. Chen (Hongkong)
F. Kamoun (Tunisia), O. Spaniol (FRG), L. Fratta (I), D. Ferrari (USA), T. L. Booth (USA)

Theory and Formalism
Vice-Chair: W. H. Kohler (USA)
P. Caspi (F), W. P. de Roever (NL), N. Francez (Israel), B. K. Bhargava (USA), R. Strong (USA)

International Liaison Committee

Chairman
Carl Davis (USA)

Vice Chairman
B. Butscher (FRG)

Secretary
Peter Ng (USA)
M. C. Chruscićki (USA)
A. Scheller (FRG)

Members:
R.-J. Back (SF)
P. van Binst (B)
B. Carlson (S)
K. Chon (Korea)
L. O. Hertzberger (NL)
P. M. Franca (Brazil)
T. Kalin (YU)
H. Kerner (A)
A. Kündig (CH)
L. Lenzini (I)
K.-S. Leong (Singapore)
K. Lindelien (N)
P. Linington (GB)
B. Mahon (L)
G. Müller (FRG)
A. Patel (IRL)
N. K. Pavlidis (GR)
J. Riera (E)
S. Sedillot (F)
V. A. Shaw (RSA)
M. Tokoro (J)
P. Villemoes (DK)
S. Zhongxiu (PR China)

Tutorial Chairmen
Earl E. Swartzlander (USA),
L. Pouzin (F)

Awards and Special Publ.
Ch. R. Vick (USA)

Secretary
R. M. Yanney

Treasurer
V. Tschammer (FRG)

Local Arrangements
P. Egloff (FRG),
H. Troy Nagle (USA)

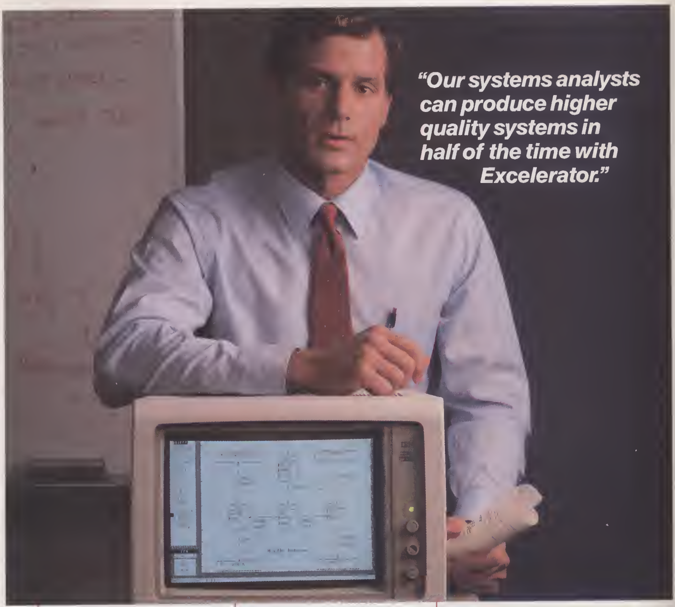
Important Dates:

January 1, 1987	submission deadline for **papers** and **tutorials**
May 30, 1987	acceptance notification

For further conference information please contact

R. Popescu-Zeletin
Hahn-Meitner-Institut
Berlin
Glienicker Straße 100
D-1000 Berlin 39 FRG

Phone
+ 49-30-80092594
2541
Network address
COMPMAIL cmp5097


"Our systems analysts can produce higher quality systems in half of the time with Excelerator."

Excelerator has established itself as the ***most comprehensive micro-based design environment*** *for systems analysts and designers. Here's why–*

1. *Mouse-driven* ***graphics*** *allow analysts to communicate project specifications clearly to both end users and programmers. Well understood requirements reduce the need for* ***costly design modifications*** *later. Excelerator supports data flow diagrams, structure charts and diagrams, data model diagrams, and presentation graphics.*

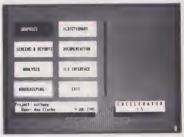
2. *Analysts* ***gain productivity*** *from having all the information about the system in one place–the Excelerator Dictionary. As your system design changes, the dictionary is updated to reflect these changes. Reviewing and updating can be done quickly and efficiently.*

3. *Analytical tools for verifying your design allow the development of* ***better systems****. Analysts can validate their designs using "where used" and "cross-reference" reporting. Explosion reports describe the data and related processes in up to 10 levels of detail. And a powerful report writer lets you produce preformatted or custom reports directly from the dictionary.*

4. *Project team members can access a central data dictionary using Excelerator's* ***data sharing facility.*** *Import/export functions allow users to move data from one workstation to another quickly and easily, and audit attributes automatically track changes to every entry.*

Find out how you can significantly increase productivity while cutting your development costs. For your free Excelerator Introductory Kit call 617 497-4473.

Reader Service Number 7



Project: softway
User: Ann Clarke 9 JUN 1985

Excelerator™

Index Technology Corporation
Five Cambridge Center
Cambridge, Massachusetts 02142